RELIMINARY DRAFT

1

DEPARTMENT OF DEFENSE

SOFTWARE MASTER PLAN

February 9, 1990

VOLUME II: BACKGROUND
(ANNEXES A-G)

Defense Acquisition Board (DAB)
Science and Technology (S&T) Committee
Software Working Group

Chairman:
Dr. George P. Millburn

This preliminary draft version is being released for public review and comment. The final draft version will be submitted for review and formal coordination by the S&T Committee and the DAB. The plan will then be submitted for final approval by the Secretary of Defense. It does not represent Department approval for any initiatives that are not presently authorized by the Secretary of Defense.

CONTENTS

ANNEX A

Current Department of Defense (DoD) Software Management Roles

February 9, 1990

CONTENTS

ANNEX A

A. Current DoD Software Management Roles

In order to identify actions required to improve the software management process, an understanding of the current software management roles of the offices and organizations within the DoD is essential. This annex lists those offices and organizations for which specific software management roles have been identified.

The format used within this annex is representative of current DoD organizational structure. The information that is provided reflects the first consolidated attempt within DoD to isolate the software specific roles of each office or organization. Those offices or organizations for which no specific software management responsibilities have been identified have been excluded from discussion.

In general, the research, development, acquisition and logistics functions are accomplished by the Military Departments and Defense Agencies. Offices within the Office of the Secretary of Defense (OSD) provide guidance and oversight for these activities. However, as evidenced by the information presented within this annex, the overall responsibility for software management is clearly fragmented across the DoD.

A.1 Office of the Under Secretary of Defense (Acquisition) (OUSD(A))

Software related responsibilities include:

— Developing policy and guidance for software acquisition programs.

— Validating software acquisition requirements.

— Chairing the Defense Acquisition Board (DAB).

A.1.1 Office of the Director of Defense Research and Engineering

Software related responsibilities include:

— Coordinating the development of software standards and the development of advanced software technology.

— Ensuring the use of appropriate advanced technology in the development of weapon systems.

A.1.1.1 Office of the Deputy Director of Defense Research and Engineering (Research & Advanced Technology) (DDDRE(R&AT))

Software related responsibilities include:

— Providing review, management oversight, policy guidance, and coordination for the science and technology programs in the Military Departments relating to all areas of software and computer technology, including computer software, computer architectures, computer languages, computer systems and equipment, artificial intelligence, robotics, and digital signal, data, and information processing.

— Reviewing, evaluating and monitoring program and project plans to ensure that planned efforts properly support program objectives.

— Assessing program deficiencies and recommending remedial actions.

— Recommending, where appropriate, coordination of programs between the Military Departments to minimize duplication of efforts and to maximize information transfer between Departments.

— Coordinating program activities that are interrelated with the OSD offices, Defense Agencies, and other government agencies as appropriate, assuring that all scientific/technical and management aspects have been properly evaluated and staffed prior to initiating or recommending action thereon.

— Providing analyses of the requirements and the program and project plans proposed by the Military Departments.

— Conducting comprehensive reviews of the programs within the Military Departments.

— Managing the DoD Ada program and coordinating implementation of the Ada policy with the DoD components.

A.1.1.2 Office of the Deputy Director of Defense Research and Engineering (Test & Evaluation)

Software related responsibilities include:

— Developing and coordinating policy and procedures for developmental test and evaluation.

— Overseeing major and designated acquisition programs.

— Applying Test and Evaluation (T&E) provisions to the software components of defense systems as well as the hardware components.

— Reviewing, evaluating and analyzing T&E plans, programs, and program execution to ensure that:

- system and mission objectives will not be impaired by improperly designed, implemented, or maintained software;
- quantitative goals and thresholds are articulated for the required technical characteristics of software components and subsystems responsible for carrying out critical mission functions;
- T&E of software achieves an appropriate balance with the hardware;
- systematic, quantitative, and objectively reportable software T&E is used to ensure that subsequent evaluations represent the software status (maturity) in the most realistic terms possible;
- a progressive approach to software T&E which provides for effective sharing of test results across life cycle phases and improves the vertical flow of information within the decision-making structures at the Military service and OSD levels is institutionalized.

A.1.1.3 Office of the Deputy Director of Defense Research and Engineering (Strategic & Theater Nuclear Forces)

Software related responsibilities include:

— Reviewing, evaluating, and monitoring all DoD research, development and acquisition programs in the mission areas of Strategic Offense, Strategic Defense, Theater Nuclear Forces, Space Launch Systems, Arms Control and Compliance, and relevant allied cooperative programs.

— Chairing the Strategic Systems Committee, which is responsible for identifying issues, and developing recommendations on major weapon system acquisition to the DAB.

— Reviewing, evaluating, and monitoring program and project plans to ensure that planned efforts properly support program objectives.

— Recommending, where appropriate, coordination of programs between the Military Departments to minimize duplication of efforts and to maximize information transfer between Departments.

— Providing analyses of the requirements and the program and project plans proposed by the Military Departments.

— Monitoring methods and procedures to ensure that unclassified but "sensitive" data and access is adequately protected.

— Reviewing costs of major acquisition programs (most of which have significant software dimensions).

— Maintaining local personal computer systems, database, and local mail.

A.1.2 Office of the Deputy Under Secretary of Defense (Industrial & International Programs)

Software related responsibilities include:

— Maintaining local personal computer systems, database, and local mail.

— Providing and maintaining technical support for export control.

— Coordinating research and development (R&D) programs with our Allies.

— Ensuring the continued viability of our defense industrial base.

A.1.3 Office of the Assistant Secretary of Defense (Command, Control, Communications, and Intelligence) (C^3I)

Software related responsibilities include:

— Developing policy and guidance for telecommunications, communications security, and computer systems security.

— Applying and integrating automated data processing (ADP) technology.

— Assessing the telecommunications and security requirements of Service and agency programs, and planned system architectures, and providing this assessment to the appropriate OSD oversight structure (DAB/C^3I systems Committee or Major Automated Information System Review Committee (MAISRC)).

A.1.4 Office of the Assistant Secretary of Defense (Production & Logistics)

Software related responsibilities include:

— Increasing interoperability through use of commercial products.

— Focusing R&D and demonstrating capabilities to improve acquisition and logistics interfaces through increased use of automation, shared data (knowledge) bases and continuing evolution of standards that allow for exchange of data.

— Developing specifications required to provide an integrated weapon system data base which incorporates digital engineering product data and logistic support data into a shared, distributed data base.

— Incorporating Computer Aided Acquisition and Logistics Support (CALS) standards and integration requirements in competitive contracts to stimulate needed industry investments.

— Developing and demonstrating CALS technology.

— Ensuring that existing infrastructure systems are modified to include CALS standards and that new systems are developed where gaps exist with the current ADP infrastructure.

— Increasing automation in areas where processes are highly rule based and repetitive and/or where critical skills are insufficient. Applying expert systems, artificial intelligence, robotics and other automation throughout the logistics systems.

— Developing methods and procedures to ensure that unclassified but "sensitive" data and access is adequately protected.

— Improving production and logistics command, control, and communications (C^3) systems.

A.2 Office of the Comptroller of the Department of Defense

Software-related responsibilities include:

— Acting as the Senior Official for Information Resources Management (IRM) and implementing 44 U.S.C. Section 3501, Public Law 96-511, "Paperwork Reduction Act of 1980," the "Paperwork Reauthorization Act of 1986."

— Ensuring that automatic data processing, automatic data processing equipment (ADPE), which includes software, telecommunications and other information technologies, are effectively and efficiently acquired and used by the Federal Government.

— Managing all software for general purpose ADP systems, except software embedded in weapons systems.

— Ensuring that policies and procedures for software higher order languages are implemented.

— For automated information systems, establishing programs, as appropriate, for the enhancement of the software engineering process and the transition of such technology from then marketplace and research programs to application within general purpose automated data processing systems.

— Promoting career development and training for personnel associated with information management activities and supporting the implementation of the DoD-wide Civilian Career Program for ADP personnel.

— Monitoring the orderly implementation of information processing standards and advanced software development and evaluation techniques.

— Establishing, issuing, and updating IRM Review Program policies and procedures and conducting periodic reviews of IRM activities including software used in connection with information processing.

— Conducting reviews of major automated information systems' (AIS) computer resources, including software, by the MAISRC of the DAB.

— Establishing, issuing, and updating End User Computing policy, including government-owned software, commercially available software packages, the development of custom software and the enforcement of software licensing provisions of the contractual vehicle used to obtain commercial software.

— Monitoring the Information Technology Users Group Program and coordinating the exchange of software information and assisting in obtaining software information from the Federal Software Exchange Center.

— Promoting and publishing standards and criteria for the interchange of software and associated documentation, consistent with the Federal Property Management Regulation (FPMR) 101-36.16 "Federal Software Exchange Program."

— Authorizing the publication of DOD-STD-7935A, "DoD Automated Information Systems (AIS) Documentation Standards."

A.3 Operational Test & Evaluation

Software related responsibilities include:

— Developing and coordinating policy and procedures for operational test and evaluation.

— Overseeing major and designated acquisition programs.

— Reviewing and approving system test and evaluation master plans and operational test plans.

— Monitoring operational tests and reporting test results to the Secretary of Defense, Under Secretary of Defense (Acquisition), and Congress.

— Testing system's effectiveness of missionized software, i.e., software that is modified or reprogrammed for each mission.

- Assessing system software suitability, including post deployment software support software, to determine if the software can be utilized and supported by the personnel designated by the using Service.

A.4 Office of the Assistant Secretary of Defense (Program Analysis and Evaluation)

Software related responsibilities include:

- Reviewing costs and effectiveness of major acquisition programs (including many with significant software dimensions) for the DAB and the Defense Review Board.
- Chairing and providing analytic support to the OSD Cost Analysis Improvement group, which provides the DAB with reports on costs of major acquisition programs at milestone review.
- Providing cost reports to OSD MAISRC at milestone reviews and chairing their Cost Effectiveness Review Group.
- Maintaining Program Analysis and Evaluation data systems.

A.5 Joint Staff

Software related responsibilities include:

- Consolidating and validating, in coordination with the Services, user requirements for applications software.
- Coordinating the integration of user requirements and information system technologies.
- Prioritizing user requirements in the competition for limited resources.
- Assessing software systems during development to assure that user requirements are being met.
- Assuring design and development address interoperability and strategic connectivity issues.
- Fostering development and implementation of multilevel security technologies.
- Staffing related manpower and funding issues.

A.6 Department of the Army

A.6.1 Office of the Assistant Secretary of the Army for Research Development and Acquisition

Software related responsibilities include:

- Ensuring that Army software technology base efforts support DoD goals.
- Focusing technology base efforts to advance the state-of-the-art in software technology (producibility and support) within the framework of the overall Army technology base program.
- Providing overall guidance and objectives to Army laboratories and research and development centers to ensure that advances in software technology support Army force modernization and preserve our advantages in this technology area.

A.6.1.1 Army Materiel Command (AMC)

A.6.1.1.1 Communications-Electronics Command (CECOM) Center for Software Engineering

Software related responsibilities include:

- Providing centralized software life cycle management engineering, and support of the Mission Critical Defense Systems (MCDS) used in Battlefield Functional Areas supported by

CECOM. These functional areas include Maneuver Control Intelligence and Electronic Warfare, Tactical Fusion, Fire Support, and Communications Systems.

— Establishing, operating and maintaining the Army Interoperability Test Bed.

— Serving as the focal point for software engineering technology that improves the productivity of the software development process. Areas include: Software process, models, methods and tools; software process metrics; software reuse, rapid prototyping and requirements engineering; software documentation; and Ada technology.

— Exploiting technology and conducting R&D focussed on improving the productivity of the software development process and thereby the quality of the software produced.

— Conducting proof of concept experimentation and demonstrating technological capabilities for developing and maintaining affordable and timely software.

— Conducting an intensive software training program for the AMC's software engineering interns.

A.6.1.1.2 CECOM Center for C^3 Systems

Software related responsibilities include:

— Researching and developing adaptable, survivable, and secure tactical information systems.

— Developing tactical decisions aids for Army commanders and their staff.

— Developing simulation systems for both tactical modeling and communications planning.

— Evaluating software systems to meet Army operational requirements. This is done with both testbeds and field exercises.

A.6.1.1.3 LABCOM

A.6.1.1.3.1 Ballistic Research Laboratory

Software related Responsibilities include:

— Investigate, develop, and evaluate innovative technology to help solve critical tactical computer problems.

— Develop Battlefield Decision Aids.

— Develop cooperating Expert Systems and a knowledge based Expert System building environment tool.

A.6.1.1.4 Army Research Office

Software related responsibilities include:

— Maintaining a basic research program component (which advances fundamental understandings enabling new software and "intelligent" system capabilities).

— Providing technical and scientific guidance upon request.

— Facilitating interaction/consultation between research and user communities (particularly through the four research centers engaged in relevant research).

A.6.1.2 Information Systems Engineering Command

Software related responsibilities include:

— Engineering,.designing, acquiring, installing, testing and accepting information for the U.S. Army, including hardware, software and system integration.

— Developing and recommending to Department of the Army for approval, hardware, software and data standards to support Army Information architectures.

A.6.1.2.1 Institute for Research in Management Information, Communication, and Computer Science

Software related responsibilities include:

- Conducting and sponsoring applied research and exploratory developments in the areas of telecommunications, automation, audiovisual, records management and publication systems which address Information Systems Command mission needs.
- Conducting transfers of existing, proven technology and providing quick-reaction technical consulting services within Information Systems Command and its supported Program Executive Officers and Program Managers.

A.6.1.3 Strategic Defense Command

Software related responsibilities include:

- Establishing requirements, and specifications for developing, testing, and demonstrating Strategic Defense Command program components.
- Planing and conducting a simulation program to resolve those critical Strategic Defense Initiative Organization (SDIO) technology issues without resorting to field testing.
- Determining, defining, and developing data processing systems hardware and software to advance ballistic missile defense capabilities.
- Maintaining cognizance of the state-of-the-art in all technical fields to insure proper and timely incorporation of new techniques and concepts into the assigned areas and expedite technology transfer.
- Establishing *configuration management, independent verification* and validation, security, quality assurance programs, Computer Resources Working Groups (CRWG) and Computer Resources Life-Cycle Management Plans (CRLCMP) to support element developments.
- Establishing, maintaining, and controlling a software repository.
- Providing software engineering training support.

A.6.1.4 Corps of Engineers

Software related responsibilities include:

- Providing life cycle development, fielding and maintenance of systems and application software to support only those business processes unique to the Corps of Engineers.
- Researching and developing applications systems required to support the missions of assigned Army Laboratory commands.

A.7 Department of the Navy (DON)

A.7.1 Office of the Assistant Secretary of the Navy (Research, Development and Acquisition) (ASN(RD&A))

Software related responsibilities include:

- Providing review, management oversight, policy guidance and coordination for Navy programs through research, development and acquisition cycles.
- Reviewing, evaluating and monitoring program and project plans to ensure that planned efforts properly support program objectives.
- Assessing program deficiencies and direct remedial actions.
- Directing, where appropriate, coordination of programs between other services to minimize duplication of efforts and maximize information transfer between Departments.

— Conducting comprehensive reviews of the programs under Navy's purview.

A.7.2 Department of the Navy, Information Resources Management (DONIRM)

The Deputy Assistant Secretary of the Navy (DASN)/ Director, Information Resources Management is the principal advisor to ASN(RD&A) on IRM issues and is charged to promote integrated policies and programs to help commanders meet information requirements through cost-effective IRM. DONIRM responsibilities include serving as the focal point for all policy and program matters relating to IRM including records management, publishing and printing, assessment, computer security, and information systems, DON senior official for mission-critical computer resources, and DON Ada executive official; overseeing information systems life cycle management; managing the DON Information System Executive Board (ISEB) and DON participation in the DoD MAISRC; procurement of computer resources through General Services Administration (GSA); and improving effectiveness of competition in ADP resource acquisition.

A.7.3 Office of the Chief of Naval Research

A.7.3.1 Office of Naval Research (ONR)

Software related responsibilities include:

— Understanding and managing complexity in algorithms, system design, and correct system construction.

— Discovering underpinnings for the design and effective use of advanced uniprocessor, parallel, and distributed computers.

— Devising theory and techniques to automate and extend human intelligence and to understand the design of intelligent, sensor-based mechanical systems.

— Creating a multi-disciplinary science base to support advanced manufacturing technologies with research thrusts in precision engineering, process modeling in shipbuilding, and reasoning about physical objects.

A.7.3.2 Office of Naval Technology (ONT)

Software related responsibilities include:

— Monitoring the commercial marketplace for COTS software products, including CASE, Computer Aided Design (CAD) and Computer Aided Engineering (CAE) tools, enabling the Navy to be a "smart buyer".

— Leveraging research investments made by other Federal/DoD agencies (such as DARPA) by developing parallel algorithms for Navy-unique problems and by exploring software solutions which use high performance computer architectures.

— Investigating selected niches in the fields of Artificial Intelligence and Artificial Neural Networks which have high pay-off potential for Navy systems.

— Providing leading-edge research relevant to interface standards and prototypes identified by SPAWAR's Next Generation Computer Resources (NGCR) program.

— Developing the specifications for generic Systems/Software Engineering Environments (SEEs) for mission-critical, software-sensitive Navy systems.

A.7.4 Office of the Chief of Naval Operations

A.7.4.1 Office of the Director of Navy Requirements for Research and Development, Test and Evaluation (OP-098)

Software related responsibilities include:

— Maintaining a basic research component.

- Providing scientific and technical guidance upon request.
- Facilitating interaction/consultation between research and user communities.
- Developing and maintaining operational test and evaluation policy and procedures for designated acquisition programs. Policy and procedure responsibilities include reviewing and/or approving system requirements documents, test and evaluation master plans and reviewing subsequent testing results, focusing on assessment of suitability and effectiveness of systems for their intended mission.
- Developing policy regarding and processing operational requirements documents, program management plans and program change approval documents.

A.7.4.2 Office of the Director of Space Command and Control (OP-094)

Software related responsibilities include:

- Overseeing DONIRM. DONIRM is responsible for policy and procedures relating to computer resources, including mission-critical and non-mission critical hardware and associated software.
- Performing requirements coordinator function for Navy standard mission-critical hardware.
- Providing waiver control authority for software languages for programs in development under Navy purview.

A.7.5 Headquarters, U.S. Marine Corps

A.7.5.1 Assistant Chief of Staff, Command, Control, Communications, Computer, Intelligence, and Interoperability (C^4I^2) Department

Software related responsibilities include:

- Establishing and overseeing Marine Corps policy for: interoperability, intraoperability, compatibility, and interfaces among command, control, communications, and intelligence (C^4I) systems; mission-critical computer resources; AIS; and communications security.
- Monitoring the development of software for tactical command and control (C^2), telecommunications, data communications, and AIS to ensure orderly progress, integration, and conformance to established standards.
- Acting as Computer Resources Support Logistics Element Manager for the Marine Corps.
- Approving developed ADP and telecommunications standards and protocols for use within the Marine Corps.
- Approving certification procedures for tactical data systems and equipment, and associated hardware.
- Developing and publishing IRM standards and guidelines.
- Developing policies and plans relating to C^4I systems architecture.
- Sponsoring the Marine Corps General Officer Computer Orientation Course.

A.7.6 Marine Corps Research, Development, and Acquisition Command

Software related responsibilities include planning and managing the acquisition of systems with mission-critical computer resources.

A.7.6.1 Marine Corps Tactical Systems Support Activity

This group provides the only Marine Corps tactical system Software Support Activity.

A.7.7 Naval Sea Systems Command

Software responsibilities include:

— Planing and directing software technology developments to satisfy operational command's needs.

— Researching, developing, testing, installing and providing life cycle support for applications software as part of Navy Weapons Systems.

— Performing technical and contractual management of Navy support software development contracts.

— Specifying new Navy software requirements to efficiently use state-of-the-art software development methods, software tools, host computers, and target computers for evolving Navy applications of embedded computer programs.

A.7.8 Space and Naval Warfare Systems Command

Software responsibilities include:

— Preparing Activity and OPR for: DOD-STD-2167A, MIL-HDBK-287, DOD-STD-2167, DOD-STD-1679A, MIL-HDBK-281, and their associated Data Item Descriptions.

— Implementing SECNAV and OPNAV instructions on computer software and security through SPAWARINST, standards, TADSTANDs and guidebooks in the SYSCOMs (NAVSEA, NAVAIR, and SPAWAR) and the Navy.

— Acting as Navy-wide OPR for RDA-MCCR and R&D computer security.

— Administering the MCCR waiver process.

— Assisting program managers in interpreting DoD and Navy standards and policies.

— Providing program managers software tools; e.g., tailoring tools for DOD-STD-2167A and Data Item Descriptions, cost-estimation models, and software reliability models.

— Performing research in computer software engineering and computer security engineering.

— Performing research in the validation of software metrics.

— Being a member of the JLC committees and subgroups. Developing standards and guidelines on software DoD-wide.

— Initiating a "DOD-STD-SYSTEM" which would be a standard for developing a system and would provide ports for hardware, software, human engineering, environmental effects, security, safety, RMA, quality (MIL-Q-9858), etc.

— Participating in several IEEE and ISO software standardization activities. Currently editor of the upcoming ISO Standard on Software Life Cycle Process.

A.8 Department of the Air Force

A.8.1 Assistant Chief of Staff, Systems for Command, Control, Communications and Computers (C^4)

Software related responsibilities include developing Air Force policy on the acquisition, programming, management and use of systems for C^4 and serving as the Air Force Ada Executive Official.

A.8.2 Office of Assistant Secretary of the Air Force (Acquisition)

The Assistant Secretary of the Air Force (Acquisition) is responsible for evaluation of Science and Technology (S&T) software program/project plans, providing Air Force representative to the DAB S&T Committee. In addition, this office is responsible for oversight of C^4 programs and formulation and development of C^4 policy. Specifically, the software responsibilities of the Deputy Assistant Secretary include:

— Establishing Air Force policy for C^4 including mission critical and embedded computer systems acquisition and management, and the development of new software environments and capabilities.

— Providing oversight of C^4 programs. Oversight includes program planning, participating in the formulation and development of program definition and monitoring progress of C^4 programs.

— Reviewing and coordinating on long term investment strategies for C^4, mission area plans and roadmaps, and acquisition strategies and plans.

— Representing the Air Force on the appropriate DAB Committee(s).

— Preparing the Acquisition Executive, jointly with the cognizant Director, for DAB meetings.

— Participating, as appropriate, on Business Strategy Panels, Source Selection Advisory Councils, or other groups, panels, boards, and committees.

— Coordinating and facilitating communications with OSD and other agencies on C^4 programs and policy matters.

— Implementing the Air Force Information Resource Management Program in accordance with P.L. 96-511, the Paperwork Reduction Act, DoD Directive 7740.1, and SAFO 560.1.

— Working in concert with the Administrative Assistant to the Secretary of the Air Force, who is responsible for the functions associated with the collection, creation, use and dissemination of information.

— Administering the Automated Information System Acquisition Review Council and its activities.

— Providing the Assistant Secretary (Acquisition) independent assessments.

— Conducting other functions directed by the Assistant Secretary (Acquisition).

A.8.3 Air Force Systems Command (AFSC)

Software related responsibilities include:

— Planning and directing software technology developments to satisfy operational commands' needs.

— Assisting Major Air Commands to define requirements for future weapons systems and assessing software technology applications.

— Formulating system concept developments to demonstrate the utility and effectiveness of software technology alternatives.

A.8.3.1 Wright Research and Development Center (WRDC)

The WRDC is responsible for the development of application software for avionics subsystems, flight control systems, propulsion systems and materials research support software. Research is centered on real time embedded flight software issues of methodology, producability, fault tolerance, testability, enhancability, instrumentability, adaptability and supportability. Goals of the research are to improve supportability, reliability, producibility, and performance of mission critical and flight safety software during their development and post-development cycles. WRDC/Avionics Lab serves as the Mission Critical Computer Resources (MCCR) focal point for WRDC, as the focal point for 6.2 software research and as the AFSC lead for the Embedded Computer Resources Support Improvement Program, a joint AFSC/AFLC program. AFSC is responsible for the identification of software support and software availability issues and the development of software tools, environments and methodologies to address these issues.

A.8.3.2 Rome Air Development Center (RADC)

RADC is chartered with developing and exploiting generic computing technologies for the Air Force. From a software perspective, the technologies currently pursued under this umbrella include software engineering methods and tools, artificial intelligence, distributed systems and data bases, and computer security. RADC is also tasked with developing and demonstrating these technologies along applications which are unique to or most critical to the Air Force C^3I communities. The current RADC software program has significant thrusts in Software and Systems Requirements Engineering and Prototyping; Software and System Quality; Software Engineering Environments and Tools; Software Engineering for Parallel and Distributed Systems; Knowledge-based Planning; knowledge-based System Engineering; Knowledge-based Software Assistant; Distributed Operating Systems; Parallel Algorithms; Distributed Database Management Systems; Real-time and AI-based Data Architectures; Computer Security Modeling; Multilevel Secure Database Management; Secure Distributed Processing; Formal Verification/Assurance; and Certification Technology.

A.8.3.3 Air Force Armament Laboratory (AFATL)

AFATL is responsible for performing software R&D for the armament domain and managing the Ada 9X development effort.

A.8.3.4 Space Technology Center

The Space Technology Center is responsible for performing software R&D for the space domain.

A.8.3.5 Human Resources Laboratory

The Human Resources Laboratory is responsible for performing software R&D for the human interface domain.

A.8.4 Air Force Logistics Command (AFLC)

AFLC is charged with the post deployment software support for the fielded weapon systems. As such, millions of lines of code are modified and enhanced at the Air Logistics Centers (ALC). AFLC is also responsible for the support of the Avionics Integration Support Facilities, the pilot training systems support and the Automatic Test Equipment; each of which contain large amounts of software. The availability (rapid turnaround) of the software is a key concern of the ALCs. As such they initiated the Embedded Computer Resources Support Improvement Program (ESIP) to address the software problem. Tasks within the ESIP are partitioned throughout AFLC. Areas covered by the ESIP are Extendable Integration Support, Configuration Management, Software Control Center, Software Technology Support Center, Readiness, and a research and development task which is led by AFSC/WRDC/Avionics Lab.

A.9 Defense Agencies

A.9.1 Defense Advanced Research Projects Agency (DARPA)

DARPA is the corporate R&D arm of Defense. In this role, it addresses technology opportunities that respond to long-term DoD needs. In computing technology, DARPA invests in R&D projects in industry, universities, and government laboratories to address a broad range of technology opportunities. Emphasis is placed on enabling and accelerating transition of promising technologies into Defense applications. Areas of investment include computer architecture, including parallel and distributed systems, embedable high performance systems, and associated systems software; high performance networking technology and protocols; artificial intelligence, including speech and vision; microelectronics, including design tools, prototyping, and packaging; prototype applications of advanced computing technology; advanced battle simulation; robotics; and software technology. In the software technology area, DARPA is responsible for the Software Engineering Institute (SEI), the Software Technology for Adaptable, Reliable Systems (STARS) program, and a technology base program.

A.9.2 Defense Communications Agency (DCA)

Software related responsibilities include:

— Planning, developing, and supporting command, control, communications, and information systems that serve the needs of the National Command Authorities under all conditions of peace and war.

— Providing guidance and support on technical and operational C^3, and information systems issues affecting the OSD, the Military Departments, the Joint Chiefs of Staff and the Joint Staff, the Unified and Specified Commands, and the Defense Agencies.

— Ensuring the interoperability of the Worldwide Military Command and Control System, the Defense Communications System, theater and tactical C^2 systems, North Atlantic Treaty Organization and/or allied C^3 systems, and those national and/or international commercial systems that affect the DCA mission.

— Supporting national security emergency preparedness telecommunications functions of the National Communications System.

A.9.3 Defense Logistics Agency

DLA provides common logistics support to DoD services and agencies. Software is managed centrally for all DLA installations and interfaces via communications networks with service and agency systems; DLA serves as the coordinator to ensure interoperability and provides network connectivity through DLANET. DLA software distribution role is small today but centralized distribution and management is expected to grow with increasing standardization.

A.9.4 National Security Agency (NSA)

NSA software management roles include responsibility for planning, managing, developing, supporting, and conducting research in telecommunications, computer, and information systems as required for national and tactical signals intelligence (SIGINT), communications security for all departments and agencies of the U.S. Government, and computer security for the Department of Defense.

NSA's National Computer Security Center is responsible for developing security criteria and guidelines; evaluating computer hardware and software security properties; establishing programs for computer security education, training, and awareness; and encouraging the widespread availability of trusted computer systems.

The Center is also responsible for executing the Consolidated Computer Security Program which consolidates all DoD security related research and development projects into one program with the broad mission of securing architectures, databases and networks, and promoting technology in security assurances, tools and software engineering.

A.10 Strategic Defense Initiative Organization (SDIO)

A.10.1 National Test Bed (NTB)

The NTB was established to evaluate proposed systems and key technologies for the Strategic Defense Initiative (SDI). The mission of the NTB is to conduct experiments on Strategic Defense Systems (SDS), evaluate the applicability and feasibility of new technologies and demonstrate through computer simulations the feasibility of alternative SDSs.

The NTB will interconnect Army, Navy, Air Force, National Laboratories, and Test/Demonstration facilities into a distributed network. The NTB may be thought of as a network of resources with the National Test Facility (NTF) as it's hub or central facility. This composite provides the principal resources dedicated to develop and/or support experimentations and provide analysis support. The NTB is a National Resource which draws together contractors, the military, government agencies, academics and others studying SDS issues.

The NTF is host to a set of hardware and software tools to support design, development, execution and analysis of simulations and experiments in support of SDI and other DoD programs. These tools simulate, at various levels of detail, the SDS weapons, sensors, communications systems and command center elements. These tools also include simulation framework, analysis support tools and specialized threat environment and scene generator tools. These tools are used to evaluate the various Strategic Defense systems and to conduct special studies and analysis. They support a wide variety of activities, including end-to-end simulations,

C^2 experiments, and human-in-the-loop interactive gaming exercises.

The NTB is comprised of mainframes, microcomputers, minicomputers, and a variety of workstations. The major hardware platforms currently operational at the NTF include an ELXIS parallel processor, Sun and Iris workstations, VAX computers, an IBM mainframe, and a Cray 2 supercomputer. These hardware tools are interconnected by unclassified and classified local area networks. It is a true heterogeneous hardware environment, predominately UNIX-based with a capability to host a variety of industry standard operating systems. The NTB hardware supports both real time and non real time operations and reconfigurable and flexible software architecture which is portable. Ada is the programming language of choice for NTB-developed software and the NTF has a Rational Ada software development facility.

A.10.2 Software Center

The SDS Software Center's mission is to ensure the efficient development, production, integration, and validation of trusted SDS software in the DEM/VAL and full scale development (FSD) phases of SDS development. The SDS Software Center will coordinate related software efforts with other SDIO Directorates, services, commands, participating agencies, major software initiatives, including those sponsored by the Ada Joint Program Office (AJPO), DARPA (including SEI and STARS), National Aeronautics and Space Administration (NASA), and Service Laboratories to avoid duplication and ensure cost effective implementation. The SDS Software Center will perform its mission with minimal impact to the element's software development costs and schedules.

The goal of the SDS Software Center is to address critical issues in software development throughout SDS development in the areas of software quality and trust, configuration management, security, reusability, standards, interoperability, technology, and training. The SDS Software Center will acquire and disseminate the state of the practice in all these areas. All of the above goals will promote cost containment on a rapidly growing program for the future.

The SDS Software Center has six basic functions:

— Provide working examples of software engineering environments (SEE) and configuration management system components that represent the current state-of-the-art technology. The SDS Software Center will support the element's installation of appropriate versions of the SDS Software Center SEEs at their development sites.

— Maintain a library of reusable Ada components that will be made available to SDS elements software developers. This library will also contain software development documentation. SDS Software Center will acquire software products and tools and contribute software articles such as baselined functions, documentation, reusable design and code, and test data.

— Provide software education and training to ensure consistency of SDS related software acquisition, design, development, integration, and testing efforts. Technical training will cover the SDS approach to software engineering. The technical courses are aimed at the lead technical personnel who determine the local software development methodology, and standards at their element. The SDS Software Center will provide training in all phases of software development with the main goal of producing an Ada competent development community. The programmatic training will focus on software acquisition, contractor evaluation and program management methodology, and tools.

— Provide programmatic support to government and contractor software developers. A team of experts will be available to assist program managers and software developers, both government and contractors, solve software problems unique to their project.

— The production and integration of a large amount of high quality, reliable software is critical to the success of the SDS mission. Productivity-enhancing tools and methods, now in the research domain must be transitioned into the software "manufacturing" or software production domain. The SDS Software Center will promote the Manufacturing Operations, Development and Integration Laboratory concept to ensure that software production and support technologies are being addressed for SDS.

- Assist in formulation of software development standards. The SDS Software Center will also provide support to standard development organizations (e.g., IEEE, ANSI, and Ada9X). The SDS Software Center will also provide assistance in the tailoring of existing standards when needed.

ANNEX B

Existing Policies, Standards, and Guidance Regarding Software and Systems

February 9, 1990

CONTENTS

ANNEX B

B. Existing Policies, Standards, and Guidance Regarding Software and Systems

As a result of the Defense Mangement Report to the President by the Secretary of Defense in July, 1989 [OSD89], a concerted effort is currently underway within DoD to reduce the stifling burden of self-imposed DoD acquisition regulations. To further support the objectives of the Defense Management Report, the Deputy Secretary of Defense issued a memorandum on October 4, 1989, citing guidelines for developing future DoD acquisition directives and instructions.

At present, there exists a plethora of DoD directives, instructions, standards, handbooks and implementation guidance specifically related to software and systems. This annex, which provides information on each of these elements, represents the first consolidated effort within the DoD to identify all such relevant documents and to attempt to show their relationships to each other on the basis of subject categories.

Documents cited within this annex are listed in numerical order within each section. Each entry contains the document identification, document title, date of the current version of the document, identification of any document(s) it may supersede, a brief summary description of the document, appropriate index terms by which the document may be referenced, and the Office of Primary Responsibility (OPR) that maintains the document. In order to better reflect the relationship of these documents to each other, the entire list of documents is then mapped to the subject areas defined by the index terms in Table B-1.

While the contents of this annex are clearly beneficial to current efforts initiated by the Defense Management Report, the information is also essential to the identification of actions required to *improve software management* policies. One obvious conclusion from the material presented in this annex is the need for consolidation and harmonization among these sometimes conflicting documents.

B.1 Public Laws and Executive Orders

B.1.1 P.L. 81-152

TITLE: The Federal Property and Administrative Services Act of 1949
DATE: **SUPERSEDES**:
SUMMARY: As amended (63 STAT.377), provides basic procurement and management authority, including the procurement and management of telecommunications and automatic data processing resources.
INDEX TERMS: Acquisition, Management, Automatic Data Processing (ADP)
OPR:

B.1.2 P.L. 89-306

TITLE: Procurement of ADP Resources by the Federal Government (Brooks Act) of 1965
DATE: **SUPERSEDES**:
SUMMARY: (40 U.S.C. 759), is the primary law governing Federal acquisition and management of ADP hardware, software and services.
INDEX TERMS: Acquisition, Management, Automatic Data Processing (ADP)
OPR:

B.1.3 P.L. 96-511

TITLE: The Paperwork Reduction Act of 1980
DATE: **SUPERSEDES**:
SUMMARY: (44 U.S.C. 35), establishes a broad mandate for agencies to perform their information activities in an efficient, effective, and economical manner. This Act also requires Federal agencies to select a Designated Senior Official who reports directly to the agency head, as the responsible party for information management activities. The Paperwork Reduction Act addresses and redefines management functions covered under the Federal Property and Administrative Services Act of 1949 (40 U.S.C. 759). The Paperwork Reduction Act specifically recognizes General Services Administration's (GSA) role in the acquisition and management of ADPE.
INDEX TERMS: Management
OPR:

B.1.4 P.L. 97-86

TITLE: Law Inapplicable to the Procurement of Automatic Data Processing Equipment and Services for Certain Defense Purposes (Warner Amendment of 1981)

DATE: **SUPERSEDES**:
SUMMARY: (10 U.S.C. 2315), amends the "Brooks Act" (P.L. 89-306) to exclude automatic data processing equipment and services for the direct fulfillment of a military or intelligence mission.
INDEX TERMS: Automated Data Processing (ADP), Acquisition
OPR:

B.1.5 P.L. 98-369

TITLE: The Competition in Contracting Act of 1984
DATE: **SUPERSEDES**:
SUMMARY: (40 U.S.. C. 759(h)), requires full and open competition as the primary method of procurement, except *under specific, well-defined circumstances. It also authorizes the GSA Board of Contract Appeals to decide protests* involving procurement of ADP equipment under the Brooks Act.
INDEX TERMS: Acquisition, Contracting
OPR:

B.1.6 P.L. 98-577

TITLE: The Small Business and Federal Procurement Competition Enhancement Act of 1984
DATE: **SUPERSEDES**:
SUMMARY: Provides that requirements for information resources may be set aside for award to small businesses.
INDEX TERMS: Acquisition, Contracting
OPR:

B.1.7 P.L. 99-591

TITLE: Paperwork Reduction Reauthorization Act of 1986
DATE: **SUPERSEDES**:
SUMMARY: (44 U.S.C. 35), refines the intent of the original act by defining "information resources management" as the "planning, budgeting, organizing, directing, training, promoting, controlling, and management activities associated with the burden, collection, creation, use and dissemination of information by agencies, and includes the management of information and related resources such as automatic data processing equipment." This Act combines the existing Automatic Data processing Fund and the Telecommunications Fund into a new multi-year Information Technology Fund administered by GSA.
INDEX TERMS: Management
OPR:

B.1.8 Executive Order 11717 - May 9, 1973

TITLE:
DATE: **SUPERSEDES**:
SUMMARY: Transferred the responsibility for clarifying and implementing Federal data processing policy from the Office of Management and Budget (OMB) to GSA and the responsibility for approving automatic data processing standards from OMB to the Department of Commerce. Fiscal and policy control, including the formulation of Federal data processing policies remained with OMB.
INDEX TERMS: Standards, Automatic Data Processing
OPR:

B.1.9 National Security Decision Directive 145

TITLE: National Policy on Telecommunications and Automated Information Systems Security - September 17, 1984
DATE: **SUPERSEDES**:
SUMMARY: Issued by the White House, sets the National Security Agency as the focal point for both military and civilian information security, including cryptography, telecommunications systems security, and automated systems security.
INDEX TERMS: System Security
OPR:

B.2 Federal Information Resources Management Regulations (FIRMR)

TITLE: Federal Information Resources Management Regulation (FIRMR)
DATE: **SUPERSEDES**:
SUMMARY: Software-related regulations include: FIRMR establishes (a) a single regulation for use by Federal or executive agencies (as applicable) governing their information activities regarding the management, acquisition, and use of certain automatic data processing, records, and telecommunications resources and (b) a new regulatory system consisting of the FIRMR and agency regulations that implement or supplement the FIRMR.

The General Services Administration is chartered with preparing, issuing and maintaining the FIRMR. This document is the primary regulation governing Federal agencies' management, acquisition, and use of certain ADP and telecommunications resources. The FIRMR is to be used in conjunction with general procurement and contracting regulations contained in the *Federal Acquisition Regulation (xxxFAR)*. Agency compliance with the FIRMR is the responsibility of the agency head as stated in 201-1.202 of the Paperwork Reduction Act and the Office of Federal Procurement Policy Act Amendments (41 USC 414). The FIRMR contains the policies pertaining to IRM as necessary to comply with the provisions of public laws. The FIRMR is the primary policy and procedure guideline for IRM in the Federal Government. As such, all policy and procedures issued by subordinate management levels are written for the purpose of implementing or supplementing FIRMR direction. Although all parts of the FIRMR are valuable references

for all aspects of IRM, the following are highlighted:
Part 201-2.001 - Definition of Software. In the FIRMR, "Software" means computer programs, procedures, rules or routines specifically designed to make use of and extent the capabilities of ADPE and includes operating systems, assemblers, compilers, interpreters, data base management systems, utility programs, sort-merge programs, maintenance diagnostic programs, and applications programs. The term encompasses operating systems software, independent subroutines, related groups of routines, sets or systems of programs, software documentation, firmware and computer data bases whether Government owned or commercially available.
Part 201-13.000 - This part prescribes policies and procedures pertaining to standards and other aspects of information resources management. The FIRMR implements two types, (1) the Federal Information Processing Standards (FIPS) and (2) Federal Telecommunications Standards (FED-STD), and seven categories of Federal Standards. The Federal ADP and Telecommunications Standards Index contains relevant information about the Federal standards. Software standards include areas of standardization such as programming languages, operating systems, operating procedures, and documentation.
Part 201-16 - Planning and Budgeting for Information Resources Activities. This part requires agencies to prepare and submit annual agency wide ADP plans in accordance with OMB Circular A-11. A copy of this plan shall be provided to GSA concurrently with each submission to OMB, as well as other information.
Part 201-20 - ADP Management Programs. This part prescribes policies and procedures regarding administrative programs for the planning, organizing, and controlling of resources for agency ADP requirements.
Part 201-21 - Telecommunications Management Programs. This part prescribes policies and procedures regarding administrative programs for the planning, organizing, and controlling of resources for agency telecommunications requirements.
Part 201-26 - Reporting Requirements. This part lists reporting or data submission requirements described elsewhere in the FIRMR regarding specific functional activities or specific subjects.
Appendices. This part contains temporary regulations, bulletins and all current issuances of documents pertaining to IRM. Appendix C contains a listing of bulletins, handbooks, and guides that pertain to the entire IRM subject area.
INDEX TERMS: Management, Acquisition, Computer Resources, Standards
OPR:

B.3 Acquisition Regulations

B.3.1 FAR

TITLE: Federal Acquisition Regulation (FAR)
DATE: **SUPERSEDES**:
SUMMARY: The FAR contains general acquisition regulations for Federal procurements. Acquisition regulations of the FIRMR are the special category of procurement and contracting regulations to be used in conjunction with FAR. Contracting for certain ADP and telecommunications resources shall be accomplished in accordance with the FAR, the FIRMR, and The DoD FAR Supplement.
— 31.001 - Defines Automatic Data Processing Equipment (ADPE)
— 31.205-2 - Addresses Automatic Data Processing Equipment Leasing Costs
INDEX TERMS: Acquisition, Contracting
OPR:

B.3.2 DoD Federal Acquisition Regulation Supplement (DFAR)

TITLE: DoD Federal Acquisition Regulation Supplement (DFAR)
DATE: **SUPERSEDES**:
SUMMARY: Software-related regulations include:
Part 227.4 - Technical Data, Other Data, Computer Software, and Copyrights. This subpart sets forth the Department of Defense policies, procedures, implementing instructions, solicitation provisions, and contract clauses relating to requirements for the acquisition of technical data and computer software as well as rights in technical data, other data, computer software, and copyrights. These sections do not encompass rights in computer software acquired under GSA authorized ADP Schedule Pricelist contracts. Such rights are governed by the terms of the GSA contracts.
Part 270 - Acquisition of Computer Resources. This part prescribes the policies and procedures for DoD and its contractors' acquisition of computer, computer components, software, maintenance services, certain other services, and computer supplies. Separate subparts address acquisition when the procurement authority is vested in the GSA, or is within the provisions of 10 USC 2315 (Warner Amendment), or when the acquisition does not fall within the scope of either GSA or 10 USC 2315 authority such as software.
270.1 - GENERAL
270.101 Procurement Authority
Software. The authority to acquire commercially available software is vested in the General Services Administration and shall be acquired in accordance with Subpart 270.3 unless within the scope of 10 USC 2315 (Warner Amendment), in which case the commercially available software shall be acquired in accordance with Subpart 270.4.
270.2 - Definitions
270.3 - Acquisition Under General Service Administration (GSA) Authority. This Subpart sets forth policies and procedures for contracting by the Department of Defense for automatic data processing equipment, commercially available software, maintenance services, and certain other services and supplies, when procurement authority is vested in the GSA.
270.318 - Software Conversion Studies. Software conversion studies are performed to ensure that the user's needs are met at the lowest overall cost, price and other factors considered.
270.4 - Acquisition Under 10 USC 2315 Authority. This Subpart is applicable to acquisition of automatic data processing

equipment or services if the function, operation, or use involves, as its primary purpose, one or more the following:

a. *Intelligence Systems*. Computer resources for the Intelligence community for the research and development of, or use in, its intelligence activities.

b. *Cryptologic Systems Related to National Security*. Computer resources for the research and development of, or use in, cryptologic activities authorized by the National Security Agency.

c. *Command and Control of Military Forces*. Computer resources for the research and development of, or use in:
   1. The National Military Command System;
   2. Worldwide Military Command and Control System;
   3. DoD Component Command and Control Systems.

d. *Integral Part of a Weapon System*. Computer Resources:
   1. Physically a part of, dedicated to, or essential in real time to, performance of the mission of weapon systems;
   2. Used for specialized training, diagnostic testing and maintenance, simulation, or calibration of weapons systems;
   3. Used for research and development of weapons systems;
   4. Used for research and development of weapons systems.

e. *Critical to the Direct Fulfillment of Military or Intelligence Missions*. Computer resources in, or used in the research and development of:
   1. Systems that will deploy as mission support in a combat environment;
   2. War planning systems;
   3. Environmental systems supporting military missions, e.g., weather, oceanographic, or satellite systems;
   4. Projects the existence of which are classified;
   5. Warning, surveillance, reconnaissance and electronic warfare systems;
   6. Mapping, charting, and geodesy systems;
   7. Airlift, sealift, and port facilities systems;
   8. Military communication systems; or
   9. Logistic systems which provide direct support to operating forces or provide direct support to maintenance of weapon systems (e.g., organic supply, software support facilities for weapon systems, etc.). This does not include logistic systems supporting, contracting, accounting, disbursement and budgeting, etc.

270.5 - *ACQUISITION UNDER OTHER AUTHORITIES*

270.6 - *ACQUISITION OF ADPE BY DoD CONTRACTORS*

270.7 - *TELECOMMUNICATIONS RESOURCES*

270.8 - *USE OF THE GSA TELEPROCESSING SERVICES PROGRAM (TSP)*

270.9 - *PRIVACY FOR COMPUTER SYSTEMS*

270.10 - *SECURITY FOR COMPUTER SYSTEMS*

270.11 - *STANDARDS*

This subpart provides for implementation of Federal Information Processing Standards (FIPS), Federal Telecommunications Standards (FED-STD) and joint FIPS/FED-STD for computer equipment subject to Subpart 270.3. These standards may also be used for computer equipment subject to Subpart 270.4 and 270.5.

270.12 - *THE FEDERAL COMPUTER PERFORMANCE EVALUATION AND SIMULATION CENTER*

270.13 - *SHARING OF COMPUTER RESOURCES*

This subpart addresses the Federal Software Exchange Program. The program applies to common use software developed or revised by either Government or contractor personnel. It is not applicable to software that is classified, developed at private expense, or developed with revolving funds where reimbursement of all costs is required. It is not applicable to software to which the Government does not possess full rights of ownership.

270.14 - *REUSE OF COMPUTER EQUIPMENT*

INDEX TERMS: Acquisition, Computer Resources, Data Rights, Security, Standards, Automatic Data Processing (ADP), Mission Critical Computer Resources (MCCR)

OPR:

B.4 DoD Policy and Guidance

This section provides a synopsis of DoD Directives, DoD Instructions, and DoD Manuals related to software and systems.

B.4.1 DoD Instruction 3235.1

TITLE: Test and Evaluation of System Reliability, Availability, and Maintainability

DATE: 01 Feb 82 **SUPERSEDES**:

SUMMARY:

INDEX TERMS: Test and Evaluation (T&E), Reliability, Maintainability

OPR: OUSD(A), H. Kimmel, 695-4421

B.4.2 DoD Handbook 3235.1-H

TITLE: Test and Evaluation of System Reliability, Availability, and Maintainability–A Primer

DATE: 01 Mar 82 **SUPERSEDES**:

SUMMARY:

INDEX TERMS: Test and Evaluation (T&E), Reliability, Maintainability

OPR: OUSD(A), H. Kimmel, 695-4421

B.4.3 DoD Directive 3405.1

TITLE: Computer Programming Language Policy
DATE: 02 Apr 1987 **SUPERSEDES**: DoD Instruction 5000.31, 24 Nov 1976
SUMMARY: Supersedes DoD Instruction 5000.31, "Interim List of DoD Approved Higher Order Programming Languages (HOL)," November 24, 1976 and establishes policy for computer programming languages used for the development and support of all DoD software. Specifies that software be acquired, based on an analysis of life-cycle costs and impact, using the following order of preference: (1) off-the-shelf application packages and advanced software technology, (2) Ada-based software and tools, and (3) approved standard HOLs. Mandates the use of Ada for Defense computer resources used in intelligence systems, for the command and control of military forces, or as an integral part of a weapon system and requires Ada for all other applications unless use of another approved standard HOL (listed in the enclosure) is more cost-effective over the life-cycle of the application. Defines a waiver process and designates language-control agents for each approved language.
INDEX TERMS: Ada Programming Language, Software Development, High Order Programming Language (HOL), Weapon Systems,
OPR: Office of DoD Comptroller, R. Harney, 697-8636

B.4.4 DoD Directive 3405.2

TITLE: Use of Ada in Weapons Systems
DATE: 30 Mar 1987
SUMMARY: Complements DoD Directive 3405.1, providing more detailed policy on the use of Ada in weapon systems. Mandates the use of Ada in computers integral to weapon systems and is applicable to new systems as well as major upgrades (redesign or addition of more than one-third of the software) to these systems. Requires the use of validated Ada compilers, software engineering principles that facilitate the use of Ada, and Ada-based program design languages. Requires that each Component designate an Ada Executive Official to serve as a focal point in the component for all Ada program activities. Requires each Component to develop an Ada implementation plan and defines the waiver process.
INDEX TERMS: Ada Programming Language, Weapon Systems, Mission Critical Computer Resources (MCCR), Software Development, Programming Languages
OPR: OUSD(A), J. Solomond, 694-0208

B.4.5 DoD Directive 4100.39

TITLE: The Defense Integrated Data System
DATE: 11 Sep 1980 **SUPERSEDES**: DoD Directive 4100.39, 31 Mar 1972
SUMMARY: Establishes policies, objectives, and responsibilities governing the design, development, operations, and maintenance of the Defense Integrated Data System (DIDS). States that DIDS shall be designed to establish and maintain an automated integrated data system to enhance the facilities for generation, receipt, dissemination, and disposition of item identification and item-related logistic management data throughout the DoD, other federal agencies, North Atlantic Treaty Organization (NATO) countries, and other foreign governments participating in DIDS. Establishes a DIDS management information reporting system, responsive to the needs of materiel managers, that provides the visibility needed to evaluate the progress and effectiveness of their respective materiel management programs. Provide for the issuance of DoD 4100.39-Manual, *Defense Integrated Data System (DIDS) Procedures Manual*, which prescribes the basic operating guidance, instructions, and supplemental policies for the DIDS, pursuant to DoD Directive 5025.1 (*Department of Defense Directive System*). Requires that existing DoD standard data elements shall be used for all data requirements to the greatest extent possible, in accordance with the provisions of DoD Directive 5000.11. These standard data elements are contained in DoD 5000.12-Manual.
INDEX TERMS: Defense Integrated Data System (DIDS), Data Collection, Logistics, Automated Information Systems (AIS), Data Elements

B.4.6 DoD 4100.39-Manual

TITLE: Defense Integrated Data System (DIDS) Procedures Manual
DATE: 11 Sep 1980 **SUPERSEDES**:
SUMMARY: Prescribes the basic operating guidance, instructions, and supplemental policies for the DIDS, pursuant to DoD Directives 4100.39 and 5025.1 (*Department of Defense Directive System*).
INDEX TERMS: Defense Integrated Data System (DIDS), Data Collection, Logistics, Automated Information Systems (AIS)
OPR: OASD(P&L), R. Allen, 697-3151

B.4.7 DoD Directive 4155

TITLE: Quality Program
DATE: **SUPERSEDES**:
SUMMARY: This directive requires a quality program and objectives for all items to be procured, in house and contractor. It specifically authorizes the Quality Assurance Council and DoD responsibilities in the Quality Program.
INDEX TERMS: Quality Assurance
OPR: OASD(P&L), P. Angiola, 695-7915

B.4.8 DoD Directive 4630.5

TITLE: Compatibility and Interoperability of Tactical Command, Control, Communications, and Intelligence Systems
DATE: 09 Oct 1985 **SUPERSEDES**:
SUMMARY: This directive establishes policy and procedures to ensure the interoperability and compatibility of tactical

command, control, communications and intelligence (C^3I) systems. It establishes policy for the development, acquisition, and deployment of tactical C^3 systems and provides mechanisms for the review of joint and combined service/agency and single service/agency requirements documents by the Joint Tactical Command, Control, and Communications Agency. It mandates systems testing to ensure interoperability among these systems.
INDEX TERMS: Command, Control, Communications and Intelligence (C^3I), Interoperability
OPR: OASD(C^3I), T. Parrish, 695-2855

B.4.9 DoD Directive 5000.1 (Change 1)

TITLE: Major and Non-Major Defense Acquisition Programs
DATE: 1 Sep 1987 **SUPERSEDES**: DoD Directive 5000.1, 12 Mar 1986
SUMMARY: Establishes policies, practices, and procedures to govern the acquisition of major and non-major defense acquisition programs. Requires a streamlined Component acquisition structure. Defines major defense acquisition programs as those designated as major by the Secretary of Defense because of urgency of need, development risk, joint funding, significant Congressional interest, or other considerations, or those estimated to require eventual expenditure for research, development, test and evaluation (RDT&E) of more than $200 million or an eventual procurement expenditure of more than $1 billion; major defense acquisition programs are designated as Defense Acquisition Board (DAB) programs or Component programs. Discusses milestone decision points for both major and non-major systems. Requires thorough validation of requirements, consideration of potential common-use solutions, analysis of affordability, conduct of activities to enhance program stability and tailor acquisitions, consideration of impact on the U.S. defense base, and consideration of cooperative acquisition efforts with U.S. Allies. Includes some procedures, but detailed procedures, supporting documentation requirements, and responsibilities for DAB and Component programs are provided in DoD Instruction 5000.2.
INDEX TERMS: Acquisition, Systems/Software Development
OPR: OUSD(A), D. Anderson, 695-9692

B.4.10 DoD Instruction 5000.2

TITLE: Defense Acquisition Program Procedures
DATE: 1 Sep 1987 **SUPERSEDES**: DoD Instruction 5000.2, 12 Mar 1986
SUMMARY: Sets forth uniform procedures governing major defense acquisition programs and establishes specific requirements and responsibilities for acquiring major defense acquisition programs requiring decision authority by the Secretary of Defense (Defense Acquisition Board (DAB) programs). States that these procedures should be generally employed for the management of acquisition programs not requiring decision authority by the Secretary of Defense (i.e., Component and non-major defense acquisition programs), as determined by the DoD Component Head, unless a statute prescribes specific compliance. Delineates the responsibilities of the 10 acquisition committees supporting the DAB, provides detailed descriptions of milestones, DAB procedures and documentation required such as the Mission-Need Statement, System Concept Paper, Test and Evaluation Master Plan, and Decision Coordinating Paper.
INDEX TERMS: Acquisition, Defense Acquisition Board (DAB)
OPR: OUSD(A), D. Anderson, 695-9692

B.4.11 DoD Directive 5000.3 (Change 1)

TITLE: Test and Evaluation
DATE: 28 Apr 1989 (Draft) **SUPERSEDES**: DoD Directive 5000.3, 12 Mar 1986
SUMMARY: Establishes policy and guidance for test and evaluation (T&E). Covers general policy on proper planning and budgeting for T&E, as well as specific policy concerning Developmental T&E, Operational T&E, T&E phases in the acquisition process, combined Developmental/Operational T&E, T&E for Special Acquisition Programs, T&E of computer software, T&E of system alterations and joint T&E programs. Requires that the principles and methodologies provided in DoD 5000.3-Manual-3 be applied to T&E of system computer software. Outlines the responsibilities of the Director, Operational Test and Evaluation and the Deputy Under Secretary of Defense, Research and Engineering, (Test and Evaluation).
INDEX TERMS: Test and Evaluation (T&E), Software Management, Mission Critical Computer Resources (MCCR)
OPR: OUSD(A), S. Kimmel, 695-4421

B.4.12 DoD 5000.3-Manual-1

TITLE: Test and Evaluation Master Plan (TEMP) Guidelines
DATE: October 1986 **SUPERSEDES**:
SUMMARY: Describes the format, content, and submission procedures for Test and Evaluation Master Plans (TEMPs) of major defense acquisition programs. The TEMP is the basic planning document for all test and evaluation (T&E) related to a particular system acquisition and is used by OSD and all DoD components in planning, reviewing, and approving T&E. The TEMP provides the basis for all other detailed T&E planning documents and serves as an essential element of the Defense Acquisition Board (DAB) decision-making process outlined in DoD Instruction 5000.2.
INDEX TERMS: Test and Evaluation (T&E), Test and Evaluation Master Plan (TEMP), Mission Critical Computer Resources (MCCR)
OPR: OUSD(A), S. Kimmel, 695-4421

B.4.13 DoD 5000.3-Manual-3

TITLE: Software Test and Evaluation
DATE: Nov 1987 **SUPERSEDES**:
SUMMARY: Supplements DoD 5000.3-Manual-1, *Test and Evaluation Master Plan (TEMP) Guidelines*,. It provides guidance for addressing software in the TEMP.

INDEX TERMS: Test and Evaluation (T&E), Test and Evaluation Master Plan (TEMP), Mission Critical Computer Resources (MCCR)
OPR: OUSD(A), S. Kimmel, 695-4421

B.4.14 DoD Directive 5000.11

TITLE: Data Elements and Data Codes Standardization Program
DATE: 07 Dec 1964 **SUPERSEDES**:
SUMMARY: Establishes the DoD Data Elements and Data Codes Standardization Program, and specifies related objectives, policies, and responsibilities. Objective is to facilitate data interchange and compatibility among data systems by providing for the optimum standardization and utilization of data elements and codes, through centralized guidance, control and direction.
INDEX TERMS: Data Elements, Data Codes, Standards
OPR: Office of Comptroller, T. Kurihara,, 695-4470

B.4.15 DoD Instruction 5000.12

TITLE: Data Elements and Data Codes Standardization Procedures
DATE: 27 Apr 1965 **SUPERSEDES**:
SUMMARY: Establishes policies, procedures, explanation of terms and criteria for identifying, developing, coding and maintaining standard Data Elements and their related Data Items, codes and Use Identifiers for use within the DoD. A Standard Data Manual will be published and updated for use by all concerned.
INDEX TERMS: Data Elements, Data Codes, Standards
OPR: Office of Comptroller, T. Kurihara, 695-4470

B.4.16 DoD 5000.12-Manual

TITLE: DoD Manual for Standard Data Elements
DATE: Oct 1986 **SUPERSEDES**:
SUMMARY: This manual contains the DoD data element dictionary and provides a current reference source for DoD approved standard data elements and codes. It contains procedures, instructions, guidelines and criteria for the standardization of data elements and codes.
INDEX TERMS: Data Elements, Data Codes, Standards
OPR: Washington Headquarters Service, J. Wolfe, 746-0797

B.4.17 DoD Instruction 5000.18

TITLE: Implementation of Standard Data Elements and Related Features
DATE: 17 Mar 1969 **SUPERSEDES**:
SUMMARY: This instruction established policies, procedures and schedules for implementing standard data elements and related features in DoD systems.
INDEX TERMS: Data Elements, Data Codes, Standards
OPR: Office of Comptroller, T. Kurihara,, 695-4470

B.4.18 DoD Directive 5000.29(Change 1)

(under revision)
TITLE: Management of Computer Resources in Major Defense Systems
DATE: 28 Dec 1976 **SUPERSEDES**: DoD Directive 5000.29, 26 Apr 1976
SUMMARY: Establishes policy for the management and control of computer resources (hardware and software) during the development, acquisition, deployment, and support of major Defense systems. Provides policy for managing computer resources in major Defense systems as elements or subsystems of major importance during conceptual, validation, full-scale development, production, deployment and support phases of the life cycle, with particular emphasis on computer software and its integration with the surrounding hardware. Requires appropriate requirements validation, risk analysis, configuration management, and computer resource life cycle planning.
INDEX TERMS: Acquisition, Life Cycle Management, Logistics, Mission Critical Computer Resources (MCCR), Computer Resources Life Cycle Management Plans (CRLCMPs)
OPR: OUSD(A), J. Batz, 694-0212

B.4.19 DoD Directive 5000.37

TITLE: Acquisition and Distribution of Commercial Products
DATE: 29 Sep 1978 **SUPERSEDES**:
SUMMARY: This broad policy directive requires that DoD buy commercial products to meet needs whenever possible. While software is not specifically addressed, Commercial-Off-the-Shelf (COTS) products and Non-Development Items (NDI) emphases support the policy intent.
INDEX TERMS: Acquisition, Commercial Products
OPR: OASD(P&L), G. Saunders, 695-7915

B.4.20 DoD Directive 5000.39

TITLE: Acquisition and Management of Integrated Logistic Support for Systems and Equipment
DATE: 17 Nov 1983 **SUPERSEDES**: DoD Directive 5000.39, 17 Jan 1980
SUMMARY: States policy and responsibilities for the acquisition and management of integrated logistic support (ILS) programs as an integral part of the acquisition process. Establishes the requirement for life cycle management of major system ILS and provides guidance when establishing ILS policy for less-than-major systems and equipment. Emphasizes that system readiness is as important an objective in the acquisition process as schedule and performance objectives.

INDEX TERMS: Acquisition, System Management, Integrated Logistic Support (ILS)
OPR: OASD(P&L), R. Koren, 756-8994

B.4.21 DoD Directive 5000.43

TITLE: Acquisition Streamlining
DATE: 15 Jan 1986 **SUPERSEDES**:
SUMMARY: Guidance policy relative to formulation of contract requirements; how specifications, standards and other contract requirements will be applied.
INDEX TERMS: Acquisition
OPR: OASD(P&L), F. Doherty, 553-8709

B.4.22 DoD Directive 5000.45

TITLE: Baselining of Selected Major Systems
DATE: 25 Aug 1986 **SUPERSEDES**:
SUMMARY:
INDEX TERMS: Acquisition, Baselining
OPR: OUSD(A), J. Ferrara, 695-4259

B.4.23 DoD Directive 5000.49

TITLE: Defense Acquisition Board
DATE: 01 Sep 1987 **SUPERSEDES**:
SUMMARY:
INDEX TERMS: Acquisition, Defense Acquisition Board (DAB)
OPR: OUSD(A), D. Anderson, 695-9692

B.4.24 DoD Directive 5000.52

TITLE: Defense Acquisition Education and Training Program
DATE: 22 Aug 1988 **SUPERSEDES**:
SUMMARY:
INDEX TERMS: Acquisition, Education and Training
OPR: OUSD(A), W. Wittig, 697-8334

B.4.25 DoD Directive 5000.53

TITLE: Manpower, Personnel, Training, and Safety in the Defense System Acquisition Process
DATE: 30 Dec 1988 **SUPERSEDES**:
SUMMARY:
INDEX TERMS: Acquisition, Manpower and Personnel, Education and Training, System/Software Safety
OPR: Force Management & Personnel, M. Pearse, 694-5133

B.4.26 DoD Instruction 5010.12

TITLE: DoD Technical Data Management Program
DATE: 23 Jan 1989 **SUPERSEDES**:
SUMMARY: Sets up the technical data management program, addressing the high level requirements collected from public law, DoD Federal Acquisition Regulation Supplements, etc. Included are distribution, marking, repositories, and new data item descriptions.
INDEX TERMS: Data Management, Standards
OPR: OASD(P&L), D. Langkamp, 756-2554

B.4.27 DoD Directive 5010.19

TITLE: DoD Configuration Management Program
DATE: 28 Oct 1987 **SUPERSEDES**: DoD Directive 5010.19, 1 May 1979
SUMMARY: Sets DoD policies for Configuration Management documentation and materiel including systems, equipment, subsystems, computer software, facilities, and other designated configuration items developed, acquired, operated, and supported by the Department of Defense. Authorizes development and issuance of a DoD Instruction for Configuration Management Practices and of DoD Component implementing documents, as applicable. Authorizes the establishment of a Configuration Management Advisory Group within the Production and Support Committee of the Defense Acquisition Board (DAB). States that computer hardware and software shall be specified and treated as configuration items, that appropriate configuration management techniques shall be developed for and tailored to software. Also states that firmware shall be treated under configuration management as either hardware or software, depending on the nature of post-development support.
INDEX TERMS: Configuration Management
OPR: OASD(P&L), L. Burgher, 756-2554

B.4.28 DoD Directive 5100.30

TITLE: World-Wide Military Command and Control System (WWMCCS) (Currently under revision)
DATE: 16 May 1974 **SUPERSEDES**:
SUMMARY: This directive defines the functional, organizational, and operational relationships between all WWMCCS elements and provides guidance and establishes responsibilities for the management, development, acquisition, and operation of the WWMCCS.
INDEX TERMS: Command, Control, Communications, and Intelligence (C^3I)

OPR: OASD(C^3I), W. Rathgeber, 697-7270

B.4.29 DoD Directive C-5200.5

TITLE: Communications Security (U) (currently under revision)
DATE: 06 Oct 1981 **SUPERSEDES**:
SUMMARY: Establishes policy regarding the security and protection of telecommunications systems that transmit classified and sensitive data.
INDEX TERMS: Communications, Systems/Software Security
OPR: OASD(C^3I)

B.4.30 DoD Directive S-5200.19

TITLE: Control of Compromising Emanations (U)
DATE: (under revision as draft C-5200.19)**SUPERSEDES**:
SUMMARY: Electronic telecommunications and automated information systems can produce unintentional, intelligence-bearing emanations commonly known as TEMPEST. This directive establishes a DoD TEMPEST Security Program to achieve reasonable, practical, and affordable TEMPEST security within the Department of Defense.
INDEX TERMS: Communications, System/Software Security
OPR: OASD(C^3I)

B.4.31 DoD Directive 5200.28

TITLE: Security Requirements for Automated Information Systems (AIS)
DATE: 21 Mar 1988 **SUPERSEDES**: DoD Directive 5200.28, 18 Dec 1972
SUMMARY: Establishes uniform policy for the safeguarding of classified (thereby supplementing DoD 5200.1-R. "Information Security Program Regulation," June 1986), sensitive unclassified, and unclassified information processed in AISs. Provides mandatory, minimum AIS security requirements while promoting the use of cost-effective, computer-based security features for AISs. Stresses the importance of a life-cycle management approach to implementing computer security requirements. Requires that all AISs handling classified and/or sensitive unclassified information and that require at least controlled access protection, shall implement required security features by 1992; for AISs above class C2, requires a timetable be submitted for compliance (with some exceptions). In all cases, requires accreditation prior to operation. Covers network security policy, designation of a Designated Approving Authority responsible for overall security of the AIS, access by foreign nationals to U.S. Government AIS, and use of automated means (software, firmware, or hardware) to extract non-Sensitive Compartmented Information (SCI) from an SCI system.
INDEX TERMS: Computer Security, Automated Information Systems (AIS)
OPR: OASD(C^3I)

B.4.32 DoD 5200.28-Manual (Change 1)

(under revision)
TITLE: ADP Security Manual: Techniques and Procedures for Implementing, Deactivating, Testing, and Evaluating Secure Resource Sharing ADP Systems
DATE: 25 Jun 1979 **SUPERSEDES**: DoD 5200.28-Manual, Jan 1973
SUMMARY: Provides guidelines and establishes techniques and procedures which can be used to:
(1) Implement secure resource sharing ADP systems so that with reasonable dependability, deliberate or inadvertent access to classified material by unauthorized personnel or the unauthorized manipulation of the computer and its associated peripheral devices, which could lead to the compromise of classified information can be prevented;
(2) Develop, acquire, and establish methodologies, techniques, standards, and procedures for the design, analysis, testing, evaluation, and approval of the security features for resource-sharing ADP Systems;
(3) Establish methodologies, techniques, and procedures for the physical protection of ADP Systems and components; and,
(4) Prescribe standards, criteria, and specifications for deactivating secure ADP Systems and the sanitizing of system components for disposition or utilization in unsecured environments.
Addresses the following software issues: operating system controls (separation from user mode); test and debug of application programs; system clearing, shutdown, and restart; memory protection; access controls and security labels; and terminal and user identification.
INDEX TERMS: Computer Security, Automated Information Systems (AIS), Automated Data Processing (ADP)
OPR: OASD(C^3I)

B.4.33 DoD Directive 5215.1

TITLE: Computer Security Evaluation Center
DATE: 25 Oct 1982
SUMMARY: At NSA, establishes the DoD Computer Security Evaluation Center, for the technical evaluation of computer system and network security, and related technical research with the objective of encouraging the easy availability of trusted computer systems. Establishes an Evaluated Products List. Assigns oversight of the Evaluation Center to the Under Secretary of Defense for Research and Engineering, in coordination with the Deputy Under Secretary of Defense (Policy) and the Assistant Secretary of Defense (Comptroller).
INDEX TERMS: Computer Security, Computer Security Evaluation Center , Communications
OPR: OASD(C^3I), D. Grulke, 695-7181

B.4.34 DoD Instruction 5215.2

TITLE: Computer Security Technical Vulnerability Reporting Program
DATE: 2 September 1986
SUMMARY: This Instruction establishes procedures for reporting technical vulnerabilities of automated information systems, and for the dissemination of such information to the DoD AIS community.
INDEX TERMS: Computer Security
OPR: OASD(C^3I)

B.4.35 DoD Instruction 7000.2

TITLE: Performance Measurement for Selected Acquisitions
DATE: 10 June 1977 **SUPERSEDES**:
SUMMARY:
INDEX TERMS: Acquisition, Performance Measurement
OPR: Office of Comptroller, W. Abba, 695-5166

B.4.36 DoD Instruction 7000.3

TITLE: Selected Acquisition Reports
DATE: 22 June 1987 **SUPERSEDES**:
SUMMARY:
INDEX TERMS: Acquisition
OPR: Office of Comptroller, C. Knoche, 695-5166

B.4.37 DoD Directive 7740.1

TITLE: DoD Information Resources Management Program
DATE: 20 Jun 1983 **SUPERSEDES**:
SUMMARY: Establishes the DoD Information Resources Management Program. It covers the information management activities of information technology, data elements, information collection, privacy of records, information security, statistical activities, forms, reports, and records.
INDEX TERMS: Data Elements, Data Collection, Privacy, Security, Automated Information Systems (AIS)
OPR: Office of Comptroller, D. Mullins, 695-3147

B.4.38 DoD Directive 7920.1

TITLE: Life Cycle Management of Automated Information Systems (AIS)
DATE: 20 June 1988 **SUPERSEDES**: DoD Directive 7920.1, 17 Oct 1978
SUMMARY: Establishes policy and procedures for the life cycle management of AISs. Exempts AISs integral to or embedded in a weapon system or used exclusively for cryptologic activities. States general policy to: control life cycle cost of AISs (through life cycle management review and milestone approval procedures, maximizing use of existing AISs, and maintaining management oversight layering), ensure appropriate interoperability requirements are taken into account, maximize use of advanced system design and software engineering technology, modernize existing AISs, and safeguard AIS resources. Distinguishes between major and non-major AISs (major AIS programs have anticipated program costs through deployment of over $100 million, or have estimated costs of over $25 million in any single year, or are designated as being special interest by OSD). Defines AIS life cycle management phases and milestones. States that certain AISs meeting the criteria in DoD Directive 5000.1, are governed by that directive. Requires baselining of major AIS programs in accordance with DoD Instruction 7920. 4. To improve software management, requires the use of modern software concepts, advanced software technology, software life-cycle support tools and standard programming languages in accordance with DoD Directive 3405.1. Requires the use of standards where possible with the following order of preference: FIPS and FED-STD, non-Government standards (e.g., ANSI, ISO, IEEE), and DOD-STD and MIL-STD.
INDEX TERMS: Automated Information Systems (AIS), Life Cycle Management
OPR: Office of Comptroller, T. Bozek, 697-8632

B.4.39 DoD Instruction 7920.2

TITLE: Major Automated Information System Approval Process
DATE: 20 October 1978 **SUPERSEDES**:
SUMMARY: Supplements DoD Directive 7920.1. Establishes the review and decision process and procedures for major AISs. Defines the System Decision Paper (SDP) process. Requires the preparation of an SDP following approval of the Mission Element Need Statement for review by the senior officials of the DoD Component and by OSD. Requires an updated SDP be provided to OSD at each milestone and defines the necessary additions to the SDP for each milestone. States the relationship of the major AIS approval process to the programming, planning, and budgeting system (PPBS), Program Objectives Memorandum, annual Consolidated Guidance Memorandum, and Five Year Defense Program.
INDEX TERMS: Automated Information System (AIS), Life Cycle Management, Test and Evaluation
OPR: Office of Comptroller, T. Bozek, 697-8632

B.4.40 DoD Instruction 7920.4

TITLE: Baselining of Automated Information Systems (AIS)
DATE: 21 March 1988 **SUPERSEDES**:
SUMMARY: Establishes policies and prescribes procedures for baselining AIS programs managed under DoD Directive 7920.1 and DoD Instruction 7920.2. Supplements DoD Directive 5000.45, *Baselining of Selected Major Systems*. States that all major AIS shall be baselined and that a program baseline document shall be prepared to support life-cycle

management (LCM) reviews. Defines the format of the program baseline document and the necessary additions to the document for each milestone. Provides guidelines for program managers on managing an AIS program baseline.
INDEX TERMS: Automated Information System (AIS), Baselining, Life Cycle Management
OPR: Office of Comptroller, T. Bozek, 697-8632

B.4.41 DoD Instruction 7930.1

TITLE: Information Technology Users Group Program
DATE: 25 Mar 1986 **SUPERSEDES**:
SUMMARY: Expands the focus of the DoD ADP Users Group Program and implements the users group concepts identified in the DoD End User Computing Policy, DoD Instruction 7930.5. The policy establishes an Information Technology Users Group Program to foster the exchange of ideas, information, and experience about information technology and its applications to the DoD.
INDEX TERMS: Users Group, Automated Information Systems (AIS)
OPR: Office of Comptroller, R. Harney, 697-8636

B.4.42 DoD Instruction 7930.2

TITLE: ADP Software Exchange and Release
DATE: 31 Dec 1987 **SUPERSEDES**:
SUMMARY: Establishes uniform policies for the exchange and release of ADP software. It addresses release to other Government agencies and to domestic and foreign requesters. The policy provides procedures for Federal government agencies to participate in the Federal Software Exchange Program, as required by the Federal Property Management Regulation (FPMR) 101-36.16. "Federal Software Exchange Program" FPMR 101-36.16.
INDEX TERMS: Software Exchange, Automated Information Systems (AIS)
OPR: Office of Comptroller, R. Harney, 697-8636

B.4.43 DoD Instruction 7935.1 (Change 1)

TITLE: DoD Automated Data Systems Documentation Standards
DATE: 8 Mar 1989 **SUPERSEDES**: 13 Sep 1977
SUMMARY: This instruction covers the documentation of all DoD automated information systems throughout their life cycle. It requires that DOD-STD-7935A, "DoD Automated Information Systems (AIS) Documentation Standards," will be used as the basis for documentation of all AIS projects.
INDEX TERMS: Documentation, AIS
OPR: Office of Comptroller, B. Newlin, 697-9068

B.4.44 DoD Directive 7950.1

TITLE: Automated Data Processing Resources Management
DATE: 29 Sep 1980
SUMMARY: This Directive supplements and provides policy guidance on the management and reporting of ADP resources. This directive applies to computer resources managed under DoD Directive 5000.29 only when such resources are released from dedicated operation for subsequent possible reutilization.
INDEX TERMS: Computer Resources, Automated Information Systems (AIS)
OPR: Office of Comptroller, D. Mullins, 693-3147

B.4.45 DoD 7950.1-Manual

TITLE: Defense Automated Resources Management Manual
DATE: Sep 1988
SUMMARY: This manual is authorized under DoD Directive 7950.1. The Automated Resources Management System has been implemented in the DoD under the management direction of the Defense Automated Resources Information Center in support of DoD resources managers. It provides procedures and reference material concerning three interrelated DoD efforts dealing with automation resources management: the Automation Equipment Redistribution Program, the Automation Resources Sharing Program, and the Automation Inventory Reporting Program.
INDEX TERMS: Equipment Redistribution, Automation Inventory, Program, Automated Information Systems (AIS)
OPR: Office of Comptroller, W. Beyer, 695-8701

B.5 DoD and Military Standards

This section provides a synopsis of Military Standards (including DoD Standards) related to software and systems.

B.5.1 MIL-STD-480B

TITLE: Configuration Control – Engineering Changes, Deviations and Waivers
DATE: 15 Jul 1988 **SUPERSEDES**: MIL-STD-480A, 12 Apr 1978
SUMMARY: This standard provides:
a. Requirements for maintaining configuration control of configuration items.
b. Requirements for the preparation and submission of proposed engineering changes, deviations, waivers and notices of revision.
c. Requirements for submitting the technical, fiscal and logistic supporting information necessary to define the impact of a proposed engineering change.
d. Instructions for submitting the information necessary to maintain the configuration identification in a current status.
Both MIL-STD-480B and MIL-STD-481B delineate configuration control requirements and provide instructions for preparing and submitting proposed engineering changes and related information. Of the two standards, MIL-STD-480B

covers the broader area and requires a more complete analysis of the impact if the engineering change described by an engineering change proposal (ECP) were implemented. MIL-STD-480B requires that the data package submitted with an ECP contain a description of all known interface effects and information concerning changes required in the functional, allocated, or product configuration identification. It is intended that MIL-STD-480B be imposed on prime contractors who: (1) have participated or are participating in the engineering or operational systems development of a system or high level configuration item, or (2) are being supplied with copies of the system specification and/or development specification(s), or (3) have extensive experience in the preparation of ECPs relative to high level configuration items. Such contractors have the capability of providing to the Government the majority of the information needed to properly evaluate the merits of a complex engineering change, possibly involving interrelated changes in other configuration items. MIL-STD-480B also covers requirements for submittal of deviations, waivers and notices of revisions.

INDEX TERMS: Configuration Management, Engineering Changes, Life Cycle, Program Management

OPR: NAVAL AIR SYSTEMS COMMAND
Commanding Officer
Naval Air Engineering Center
Systems Engineering and Standardization Department (SESD) Code 53
ATTN: Mr. C. Meade
Lakehurst, NJ 08733-5100
(201) 323-2326, 2621

B.5.2 MIL-STD-481B

TITLE: Configuration Control-Engineering Changes, Deviations, and Waivers (Short Form)

DATE: 15 Jul 1988 **SUPERSEDES**: MIL-STD-481A, 18 Oct 1972

SUMMARY: MIL-STD-481B is intended for use in contracts involving the procurement of multi-application items or items for which the prescribed detail design was not developed by the contractor. It sets forth requirements for the preparation and submittal of abbreviated Engineering Change Proposals (ECP). The information to be submitted emphasizes the impact on the item under contract, with a very limited description of the effect on interfaces and Integrated Logistic Support (ILS).

INDEX TERMS: Configuration Management, Engineering Changes, Life Cycle, Program Management

OPR: NAVAL AIR SYSTEMS COMMAND
Commanding Officer
Naval Air Engineering Center
Systems Engineering and Standardization Department (SESD) Code 53
ATTN: Mr. C. Meade
Lakehurst, NJ 08733-5100
(201) 323-2326, 2621

B.5.3 MIL-STD-482A

(Under Revision)

TITLE: Configuration Status Accounting Data Elements and Related Features

DATE: 1 Apr 1974

SUMMARY: This standard establishes the data elements, and their related data items, codes, use identifiers, and data chains (referred to as "related features") to be used as the content of configuration status accounting records. This standard does not prescribe which of the data elements and related features to use, the status-accounting-record format to be used, or the frequency of status-accounting-record reports. Such requirements are specified elsewhere by the procuring activity. If data elements, supplemental to those in this standard, are required by the government or a contractor for configuration management, it will be necessary to submit these data elements and related features to the custodian of this standard for tentative approval prior to use.

INDEX TERMS: Data Elements, Configuration Management, Life Cycle Management

OPR: NAVAL SEA SYSTEMS COMMAND (ORDINANCE SYSTEMS)
Commanding Officer Naval Ordinance Station
Standardization Branch (Code 3730)
Indian Head, MD 20640-5000
(301)743-4250, 4358, 4510 4723

B.5.4 MIL-STD-483A (USAF)

TITLE: Configuration Management Practices for Systems, Equipment, Munitions, and Computer Programs

DATE: 04 Jun 1985 **SUPERSEDES**: MIL-STD-483, 31 Dec 1970

SUMMARY: This standard establishes requirements for configuration management in the following areas:

a. Configuration management plan
b. Configuration identification
c. Configuration control
d. Configuration audits
e. Interface control
f. Engineering release control
g. Configuration management reports/records

Its purpose is to establish uniform configuration management practices that can be tailored to all systems and configuration items, including those systems and configuration items procured by the Air Force for other agencies.

INDEX TERMS: Configuration Management, Interface Control, System/Software Engineering

OPR: USAF

B.5.5 MIL-STD-490A

TITLE: Specification Practices
DATE: 04 Jun 1985 **SUPERSEDES**: MIL-STD-490, 30 Oct 1968
SUMMARY: This standard establishes the format and contents of specifications for program-peculiar configuration items, processes, and materials. The purpose of this standard is to establish uniform practices for specification preparation, to ensure the inclusion of essential requirements, and to aid in the use and analysis of specification content. Specifications covered by this standard may be prepared as military, Federal, contracting agency, or contractor specifications. These types of specifications are:

a. System/Segment Specification
b. Development Specifications
c. Product Specifications
d. Process Specification
e. Material Specification.

INDEX TERMS: Specifications
OPR: COMMAND STANDARDIZATION OFFICE (ComSO), HQ AFSC
HQ AFSC/PLEQ
Andrews AFB, MD 20334-5000
(301)981-2751

B.5.6 MIL-STD-499A (USAF)

TITLE: Engineering Management
DATE: 01 May 1974 **SUPERSEDES**: MIL-STD-499 (USAF) 17 Jul 1969
SUMMARY: MIL-STD-499A (USAF) has been developed to assist Government and contractor personnel in defining the system engineering effort in support of defense acquisition programs. This standard applies to internal DoD system engineering as well as joint Government-industry applications for Government contracts. The term "contractor", as used throughout this Standard, also means "government agency" when acquisition is being done in-house. The fundamental concept of this Standard is to present a single set of criteria against which all may propose their individual internal procedures as a means of satisfying engineering requirements. Economy is thus achieved by permitting a contractor's internal procedures to be used in support of Air Force programs. In those cases where multi-associate contractors are involved or when more specific direction to a contractor is essential, as determined by the program manager, a set of specific engineering task statements tailored to the specific needs of the program may be specified in the Request for Proposal (RFP).
INDEX TERMS: Program Management, Life Cycle, Acquisition, System/Software Engineering
OPR: COMMAND STANDARDIZATION OFFICE (ComSO), HQ AFSC
HQ AFSC/PLEQ
Andrews AFB, MD 20334-5000
(301)981-2751

B.5.7 MIL-STD-680

TITLE: Contractor Standardization Plans and Management
DATE: April 1971 **SUPERSEDES**:
SUMMARY: Provides background and procedural requirements for standardization programs which ensure that the use of nonstandard parts, materials, and processes is kept to an absolute minimum. This document is to furnish guidance on a standardization program for the deliverable system.
INDEX TERMS: Standards
OPR: NAVAL AIR SYSTEMS COMMAND
Commanding Officer
Naval Air Engineering Center
Systems Engineering and Standardization Department (SESD) Code 53
ATTN: Mr. C. Meade
Lakehurst, NJ 08733-5100
(201)323-2326, 2621

B.5.8 MIL-STD-881A

TITLE: Work Breakdown Structures for Defense Material Items
DATE: 25 Apr 75 **SUPERSEDES**: MIL-STD-881, 01 Nov 1968
SUMMARY: Contains work breakdown structures which are defined elements of work to structure a complete weapons system and its logistics into a numeric outline with which to budget and accumulate costs into meaningful categories. The structure lists the standard elements and defines appropriate rules for the inclusion of work activities in each element. This provides a consistent basis for accumulating all costs without omission or redundancy. MIL-STD-881A is being revised by a DoD Working Group. Over the years, software has become an increasingly significant part of the development of major acquisition systems. The MIL-STD-881A does not explicitly deal with software, although in OSD s review of each major system's WBS prior to the start of Full Scale Development every effort is taken to ensure high risk and high cost software development and support efforts are identified and cost reporting requirements are established.
INDEX TERMS: Work Breakdown Structures, Weapon Systems, Program Management
OPR: COMMAND STANDARDIZATION OFFICE (ComSO), HQ AFSC

HQ AFSC/PLEQ
Andrews AFB, MD 20334-5000
(301)981-2751

B.5.9 MIL-STD-882A

TITLE: System Safety Program Requirements
DATE: 01 Jul 1977 **SUPERSEDES**:
SUMMARY: Provides guidance for contractors to establish system safety as a formal program supporting the design and testing of a system. Content requirements for a System Safety Program Plan are furnished. This standard applies to DoD systems and facilities including test, maintenance and support, and training equipment. It applies to all activities of the system life cycle; e.g., research, design, technology development, test and evaluation, production, construction, operation and support, modification and disposal. The requirement will also be applied to DoD in-house programs.
INDEX TERMS: Safety, Requirements Specification
OPR: COMMAND STANDARDIZATION OFFICE (ComSO), HQ AFSC
HQ AFSC/PLEQ
Andrews AFB, MD 20334-5000
(301)981-2751

B.5.10 MIL-STD-1388-1A

TITLE: Logistic Support Analysis
DATE: 11 Apr 1983 **SUPERSEDES**: MIL-STD-1388-1, 15 Oct 1973
SUMMARY: This standard implements the Logistic Support Analysis (LSA) guidelines and requirements established by DoD Instruction 5000.2, Major System Acquisition Procedures, and DoD Directive 5000.39, Acquisition and Management of Integrated Logistic Support for Systems and Equipment. The requirements of this standard are applicable to major and less-than-major system/equipment acquisition programs, major modification programs, and applicable research and development projects. The goal of this standard is a single, uniform approach by the Military Services for conducting those activities necessary to (a) cause supportability requirements to be an integral part of system requirements and design, (b) define support requirements that are optimally related to the design and to each other, (c) define the required support during the operational phase, and (d) prepare attendant data products. LSA is the selective application of scientific and engineering efforts undertaken during the acquisition process, as part of the system engineering and design process, to assist in complying with supportability and other Integrated Logistic Support (ILS) objectives through the use of an iterative process of definition, synthesis, tradeoff, test, and evaluation.
INDEX TERMS: Logistics, Acquisition, Integrated Logistics Support (ILS)
OPR: AMC MATERIEL READINESS SUPPORT ACTIVITY
Commander, US AMC Materiel Readiness Support Activity
ATTN: AMXMD-MP
Lexington, KY 40511-5101
(606)293-3415

B.5.11 MIL-STD-1388-2A

TITLE: DoD Requirements for a Logistic Support Analysis Record
DATE: 14 Feb 1986, Notice 2 - 15 Jan 1987**SUPERSEDES**: MIL-STD-1388-2, 15 Oct 1973
SUMMARY: This standard prescribes the data element definitions, data field lengths, and data entry requirements for logistic support analysis record (LSAR) data. It identifies the LSAR reports that are generated from the LSAR data and identifies the input formats for the Joint Service LSAR ADP system when it is used. This standard applies to all system/equipment acquisition programs, major modification programs, and applicable research and development projects through all phases of the system/equipment life cycle.
INDEX TERMS: Logistics, Automatic Data Processing (ADP), Integrated Logistics Support (ILS)
OPR: AMC MATERIEL READINESS SUPPORT ACTIVITY
Commander, US AMC Materiel Readiness Support Activity
ATTN: AMXMD-MP
Lexington, KY 40511-5101
(606)293-3415

B.5.12 MIL-STD-1397A

TITLE: Input/Output Interfaces, Standard Digital Data, Navy Systems
DATE: 7 Jan 1983
SUMMARY: This standard describes generic functional, physical, and electrical characteristics of interfaces for Navy digital equipment, including Naval Tactical Data Systems (NTDS). It does not describe the specific interface philosophy for a given application, but is limited to functional characteristics of the interface signals. It contains data transmission rates, communication protocols, and electrical characteristics for NTDS (slow), NTDS (fast), A-NEW, NTDS Serial, and NATO Serial Interfaces.
INDEX TERMS: Interface Signals, Standards
OPR: NAVAL SEA SYSTEMS COMMAND (SHIP SYSTEMS)
Commander, Naval Sea Systems Command (SEA 55Z3)
DoD Standardization Program and Documents Division
Department of the Navy
Washington, DC 20362-5101

(202)692-0160, 0161

B.5.13 MIL-STD-1399C

TITLE: Interface Standard Shipboard Systems
DATE: 2 Feb 1988 **SUPERSEDES**: DOD-STD-1399B, 22 Nov 1977
SUMMARY: This standard identifies the data process environment for shipboard systems in terms of standards for controlled and uncontrolled environmental conditions, functional information (signals) and controlled factors relating to air conditioning, access, weight constraints, etc. Actual requirements are to be completed for each of the sections through dialogue between the procurement agency and the contractor/bidder
INDEX TERMS: Shipboard Systems, Data Processing, Computer Hardware, Interface Standards,
OPR: USN

B.5.14 DOD-STD-1467 (AR)

TITLE: Software Support Environment
DATE: 18 Jan 1985 **SUPERSEDES**: DOD-STD-1467, 07 Nov 1984
SUMMARY: This Standard establishes uniform minimum requirements for the contractor to define a Developmental Software Support Environment, to ensure the compatibility of this environment with a contracting activity's designated Life Cycle Software Support Environment, and to ensure the existence of a complete contracting activity life cycle software support capability for the deliverable software of the contracted effort.
INDEX TERMS: Software Support Environment
OPR: ARMAMENT MUNITIONS AND CHEMICAL COMMAND
Commander US Army Armament, Munitions and Chemical Command (AMCCOM)
US Army Armament Research, Development and Engineering Center
ATTN: SMCAR-ESC-S
Picatinny Arsenal, NJ 07806-5000
(201)724-6530, 6662, 6674

B.5.15 MIL-STD-1521B (USAF)

TITLE: Technical Reviews and Audits for Systems, Equipments, and Computer Software
DATE: 04 Jun 1985, Notice 1 - 19 Dec 1985**SUPERSEDES**: MIL-STD-1521A (USAF) 01 Jun 1976
SUMMARY: This standard prescribes the requirements for the conduct of Technical Reviews and Audits on Systems, Equipments, and Computer Software. The following technical reviews and audits shall be selected by the program manager at the appropriate phase of program development:

System Requirements Review
System Design Review
Software Specification Review
Preliminary Design Review
Critical Design Review
Test Readiness Review
Functional Configuration Audit
Physical Configuration Audit
Formal Qualification Review
Production Readiness Review

Technical Reviews and Audits defined herein shall be conducted in accordance with this standard to the extent specified in the contract clauses, Statement of Work, and the Contract Data Requirements List. The contracting agency shall tailor this standard to require only what is needed for each individual acquisition.
INDEX TERMS: Program Management, Test and Evaluation (T&E)
OPR: ELECTRONIC SYSTEMS DIVISION, AFSC
HQ ESD/PLP
Standardization Office
Hanscom AFB, MA 01731-5000
(617)377-2918

B.5.16 MIL-STD-1553B

TITLE: Aircraft Internal Time Division Command/Response Multiplex Data Bus
DATE: 21 Sep 1978, Notice 2 - 08 Sep 1986
SUMMARY: This standard describes requirements for coding, signal levels, and timing for use in synchronizing and data exchange by equipment operating in command/response on MIL-STD-1553B data bus. Details are provided for all BIT field use over this bus. The protocol will support up to 32 terminals, including the bus controller.
INDEX TERMS: Data Bus, Computer Hardware
OPR: AERONAUTICAL SYSTEMS DIVISION, AFSC
Standards Division
ASD/ENES
Wright-Patterson AFB, OH 45433-6503
(513)255-6295

B.5.17 MIL-STD-1574A (USAF)

TITLE: System Safety Program for Space and Missile Systems
DATE: 15 Aug 1979 **SUPERSEDES**: MIL-STD-1574, 15 Mar 1977

SUMMARY: Establishes administrative and technical means by which accident prevention requirements and policies are planned, managed, and implemented into the total program effort. Defines the requirements for implementation of system safety programs covering the life cycle of the system, including: design, development, test, checkout, modification, production, servicing, refurbishing, maintenance, transportation, handling, training, disposal, deployment, and normal and contingency operations. Provides a tailored application of MIL-STD-882A for space, missiles, and related systems.
INDEX TERMS: System/Software Safety, Space Systems, Weapon Systems
OPR: SPACE DIVISION, AFSC
SD/ALM
P.O. Box 92960
Los Angeles Air Force Station
Los Angeles, CA 90009-2960
(213)643-1966

B.5.18 MIL-STD-1577A

TITLE: Intercontinental Ballistic Missile Systems Training/Training Equipment Management
DATE: 31 Jul 1985 **SUPERSEDES**: MIL-STD-1577, 15 Mar 1980
SUMMARY: This standard defines the functions, roles and responsibilities of the Training and Training Equipment Division of the Human Factors Engineering and Management Team of the Intercontinental Ballistic Missiles Program Office, Air Training Command, the Using Commands, Air Force Logistics Command (AFLC), and the contractor(s) to: Identify all training equipment comprising the system training equipment package. Design; develop; fabricate; install and check out; and accept required training equipment. Develop Operations and Maintenance manuals required to operate and maintain trainers. Establish personnel and training requirements; and accomplish training and training equipment planning as defined in Appendices A and B. Establish training equipment requirements and prepare training equipment specifications as provided in Appendices C and E. Document training equipment and software per Appendices D and F.
INDEX TERMS: Education and Training
OPR: BALLISTIC MISSILES OFFICE, AFSC
Ballistic Missiles Office, AFSC
BMO/AWD
Norton AFB, CA 92409-6468
(714)382-4806

B.5.19 MIL-STD-1589C

TITLE: JOVIAL (J73)
DATE: 06 Jul 1984 **SUPERSEDES**: MIL-STD-1589B, 06 Jun 1980
SUMMARY: Describes the syntax and semantics of the JOVIAL programming language.
INDEX TERMS: Programming Languages, JOVIAL Programming Language, Weapon Systems
OPR: AERONAUTICAL SYSTEMS DIVISION, AFSC
Standards Division
ASD/ENES
Wright-Patterson AFB, OH 45433-6503
(513)255-6295

B.5.20 MIL-STD-1591

TITLE: Command, Control and Communications (C^3) System & Component Fault Diagnosis, Subsystems, Analysis/Synthesis of
DATE: 03 Jan 1977 **SUPERSEDES**:
SUMMARY: This standard establishes uniform criteria for conducting trade studies to determine the optimal design for command, control and communication system and component fault diagnosis/isolation subsystems, hereafter referred to as Fault Identification & Test Subsystems (FITS). FITS include the hardware and/or software necessary for the detection and isolation of failures.
INDEX TERMS: Fault Identification, Command, Control, Communications, and Intelligence (C^3I)
OPR: ROME AIR DEVELOPMENT CENTER, AFSC
RADC/RBE-2
Griffiss AFB, NY 13441-5700
(315)330-2101

B.5.21 MIL-STD-1700

TITLE: Data Management Program
DATE: 15 May 1986, Notice 1 - 28 Sep 1987
SUMMARY: This standard establishes requirements for a contractor's data management program. This standard is intended for use when referenced in the contract to assure the currency and timeliness of all contractually required data.
INDEX TERMS: Data Management, Acquisition
OPR: National Security Agency
Director for Telecommunications and Computer Services

B.5.22 MIL-STD-1701

TITLE: Hardware Diagnostic Test System Requirements
DATE: 10 Jun 1985

SUMMARY: This establishes the general procedures, terms and conditions governing the preparation and completion of a hardware diagnostic test system.
INDEX TERMS: Automatic Test Equipment (ATE), Computer Hardware
OPR: National Security Agency
Director for Telecommunications and Computer Services

B.5.23 MIL-STD-1750A

TITLE: Sixteen-Bit Computer Instruction Set Architecture
DATE: 02 Jul 1980, Notice 2 - 25 Feb 1988
SUMMARY: This standard consists of detailed information on machine-level instructional application for 16-bit, 32-bit, and 48-bit computer architectures as standards for use by the DoD and available for use by the other services. It defines instruction set in terms of the specific opcode and mnemonic for calling the instruction, and the resultant activity in the register(s).
INDEX TERMS: Instruction Set Architecture, Computer Hardware
OPR: AERONAUTICAL SYSTEMS DIVISION, AFSC
Standards Division
ASD/ENES
Wright-Patterson AFB, OH 45433-6503
(513)255-6295

B.5.24 MIL-STD-1753

TITLE: FORTRAN, DoD Supplement to American National Standard X3.9-1978
DATE: 09 Nov 1978
SUMMARY: This supplement contains the additional syntax and semantics for those facilities necessary to the operating needs of the DoD Departments and Agencies.
INDEX TERMS: FORTRAN, Programming Language
OPR: ASSISTANT CHIEF OF STAFF FOR COMMAND, CONTROL,
COMMUNICATIONS AND COMPUTERS, SYSTEMS HQ USAF
HQ USAF (SCTT)
ATTN: IPSC
Washington, DC 20330-5190
(202)695-0499

B.5.25 MIL-STD-1777

TITLE: Internet Protocol
DATE: **SUPERSEDES**:
SUMMARY:
INDEX TERMS: Communications Protocol, Command, Control, Communications, and Intelligence (C^3I)
OPR:

B.5.26 MIL-STD-1778

(To be replaced by FIPS 146, Government Open Systems Interconnect Profile (GOSIP))
TITLE: Transmission Control Protocol
DATE: 12 Aug 1983, Notice 1 - 26 Oct 1983
SUMMARY: This document specifies the Transmission Control Protocol, a reliable connection-oriented transport protocol for use in packet-switched communication networks and internetworks. The document includes an overview with a model of operation, a description of services offered to users, and a description of the architectural and environmental requirements. The protocol service interfaces and mechanisms are specified, using an extended state machine model.
INDEX TERMS: Communications Protocol, Command, Control, Communications, and Intelligence (C^3I)
OPR: DEFENSE COMMUNICATIONS AGENCY
Director, Defense Communications Agency (ATTN: R130)
1860 Wiehle Avenue
Reston, VA 22090
(703)437-2802

B.5.27 MIL-STD-1780

(To be replaced by FIPS 146, Government Open Systems Interconnect Profile (GOSIP))
TITLE: File Transfer Protocol
DATE: 10 May 1984
SUMMARY: This document specifies the File Transfer Protocol (FTP) which supports the transfer of file data throughout a heterogeneous host computer network. This draft standard defines the FTP's role and purpose, defines the services provided to users, and specifies the mechanisms needed to support the services.
INDEX TERMS: Communications Protocol, Communications, Command, Control, Communications, and Intelligence (C^3I)
OPR: DEFENSE COMMUNICATIONS AGENCY
Director, Defense Communications Agency (ATTN: R130)
1860 Wiehle Avenue
Reston, VA 22090

(703)437-2802

B.5.28 MIL-STD-1781

(To be replaced by FIPS 146, Government Open Systems Interconnect Profile (GOSIP))
TITLE: Simple Mail Transfer Protocol
DATE: 10 May 1984
SUMMARY: This document specifies the Simple Mail Transfer Protocol, a protocol designed to transfer mail reliably and efficiently. The document includes an introduction to the protocol with a model of operation, procedures, and specifications, including state diagrams.
INDEX TERMS: Communications Protocol, Command, Control, Communications, and Intelligence (C^3I)
OPR: DEFENSE COMMUNICATIONS AGENCY
Director, Defense Communications Agency (ATTN: R130)
1860 Wiehle Avenue
Reston, VA 22090
(703)437-2802

B.5.29 MIL-STD-1782

TITLE: TELNET Protocol
DATE: 10 May 1984
SUMMARY: This document specifies the TELNET protocol and a number of approved options. It provides a standard method of interfacing terminal devices and terminal-oriented processes to each other. It is envisioned that the protocol may also be used for terminal-terminal communication ("linking") and process-process communication (distributed computation).
INDEX TERMS: Communications Protocol, Command, Control, Communications, and Intelligence (C^3I)
OPR: DEFENSE COMMUNICATIONS AGENCY
Director, Defense Communications Agency (ATTN: R130)
1860 Wiehle Avenue
Reston, VA 22090
(703)437-2802

B.5.30 MIL-STD-1794

TITLE: Human Factors Engineering Program for Intercontinental Ballistic Missile Systems
DATE: 01 Oct 1986
SUMMARY: This standard provides task descriptions for conducting an integrated human factors engineering (HFE) program to the development and acquisition of intercontinental ballistic missile (ICBM) systems. These requirements include the work to be accomplished or subcontracted by the contractor in effecting an integrated HFE program. The tasks, as tailored, will be applied to systems, equipment, software and facilities studies, concept development, demonstration and validation, design development, and test, acquisitions and modifications. This standard is applicable to all ICBM weapon systems development. The responsibility for HFE begins with the inception of the system and continues throughout the life cycle of each system. The HFE program shall apply to all system engineering analyses, studies, concepts, systems, equipment, software, and facilities for which the contractor is assigned developmental responsibility in the contract, including aerospace vehicle equipment, operational support equipment, maintenance support equipment, depot support equipment, test support equipment, special test equipment, training equipment, and modifications to government furnished equipment and commercial equipment. This responsibility must be addressed in each system management, planning, programming or contractual document.
INDEX TERMS: Human Factors, Weapon Systems, Missile Systems
OPR: BALLISTIC MISSILES OFFICE, AFSC
Ballistic Missiles Office, AFSC
BMO/AWD
Norton AFB, CA 92409-6468
(714)382-4806

B.5.31 MIL-STD-1803 (DRAFT)

TITLE: Software Development Integrity Program (SDIP)
DATE: 8 Jun 88 **SUPERSEDES**:
SUMMARY: This standard provides general requirements and specific tasks to achieve software integrity during the development and deployment of systems and equipment. This standard, when used in conjunction with DOD-STD-2167A, forms the basis for the SDIP. The SDIP is intended to provide disciplined process for assuring that software is developed to achieve specified performance and improved supportability. The SDIP will emphasize integrity on a par with cost, schedule, and other design and performance criteria throughout the acquisition cycle. It is intended that this standard not duplicate or contradict any requirement of DOD-STD-2167A.
INDEX TERMS: Computer Resources, Software Integrity, Software Development
OPR:

B.5.32 ANSI/MIL-STD-1815A

(Under Revision)
TITLE: Ada Programming Language
DATE: 17 Feb 1983 **SUPERSEDES**: MIL-STD-1815, 22 Jan 1983
SUMMARY: This standard specifies the form and meaning of program units written in Ada. Its purpose is to promote

the portability of Ada programs to a variety of data processing systems.

INDEX TERMS: Ada Programming Language, Programming Languages, Weapon Systems, Automated Information Systems (AIS)

OPR:
OUSD(A)/DDDRE(R&AT)
Director, AJPO
Rm 3E114, Pentagon
Washington, DC 20301-3081
(202)694-0208

B.5.33 MIL-STD-1838A

TITLE: Common Ada Programming Support Environment (APSE) Interface Set (CAIS)

DATE: 06 Apr 1989 **SUPERSEDES**: DOD-STD-1838, 09 Oct 1986

SUMMARY: This document provides specifications for a set of Ada packages, with their intended semantics, which together form a set of common interfaces for Ada Programming Support Environments (APSEs). This set of interfaces is known as the Common APSE Interface Set (CAIS) and is designed to promote the source-level portability of Ada programs, particularly Ada software development tools. The CAIS as specified in this document is believed to provide the tool writer with significant advantages in tool portability, since the most crucial host dependencies are transparently encapsulated by the interfaces of the CAIS. The CAIS is not a replacement for a host operating system. It standardizes those tool-to-host interfaces that are most crucial for tool portability. Other less frequently used or inherently host-dependent interfaces must complement the CAIS in order to provide a basis for the construction of an APSE. The degree of portability of tools will depend on the degree to which they can obtain the required functionality of host interfaces through the CAIS, rather than through host-dependent interfaces.

INDEX TERMS: Ada Programming Language, Ada Programming Support Environment (APSE), Portability

OPR:
ASSISTANT CHIEF OF STAFF, SYSTEMS FOR COMMAND, CONTROL,
COMMUNICATIONS AND COMPUTERS,
HQ USAF (SCTT)
ATTN: IPSC
Washington, DC 20330-5190
(202)695-9935

B.5.34 MIL-STD-1840AOPR:

TITLE: Computer Aided Acquisition and Logistic Support (CALS) Data Exchange Specification

DATE::

SUMMARY: This standard defines file exchange formats *(headers / envelopes) and processes for CALS information* transfer between government and industry across various media including quality measures.

INDEX TERMS: Data Code, Logistics, Metrics, Interface Control

OPR: OASD(P&L) Bruce Lepisto 697-0051

B.5.35 MIL-STD-1862B

TITLE: Nebula Instruction Set Architectures

DATE: 03 Jan 1983 **SUPERSEDES**: MIL-STD-1862A, 02 Nov 1981

SUMMARY: This standard defines the Nebula Instruction Set Architecture. The instruction set includes information for writing any time-dependent program that will execute on computers conforming to this standard. This is an independent architecture not representative of any particular computer.

INDEX TERMS: Instruction Set Architecture, Nebula

OPR:
COMMUNICATIONS - ELECTRONICS COMMAND
Commander, US Army Communications Electronics Command and Fort Monmouth
ATTN: AMSEL-ED-TO
Fort Monmouth, NJ 07703-5000
(201)532-5851, 5852, 5853, 5854

B.5.36 MIL-STD-2001

TITLE: Manuals, Technical, Systems Operator's Interface: Procedures for Writing

DATE: 14 Jan 1987 **SUPERSEDES**:

SUMMARY: This standard establishes uniform style and format requirements for preparation of Systems Operator's Interface Technical Manuals. It also specifies all electrical, mechanical, software, and Man Machine Interface (MMI) requirements necessary to provide an operational communications network.

INDEX TERMS: Documentation, Communications

OPR:
UNITED STATES MARINE CORPS
Commanding General, Marine Corps Research, Development, and Acquisition Command,
(Code PSE-C)
Washington, DC 20380-0001
(202)694-2606, 1341

B.5.37 MIL-STD-2076

TITLE: Unit Under Test Compatibility with Automatic Test Equipment, General Requirements for

DATE: 01 Mar 1978 **SUPERSEDES**: AR-8B, 24 Dec 1974

SUMMARY: This standard contains requirements for design features which must be incorporated into equipment which is to be supported with Automatic Test Equipment (ATE). It includes those necessary electrical and mechanical attributes, as well as access needs which will facilitate full, effective and efficient utilization of ATE to perform test and diagnostic functions.

INDEX TERMS: Test Specifications, Automatic Test Equipment (ATE), Weapon Systems

OPR:
NAVAL AIR SYSTEMS COMMAND
Commanding Officer, Naval Air Engineering Center
Systems Engineering and Standardization Department (SESD) Code 53
ATTN: Mr. C. Meade
Lakehurst, NJ 08733-5100
(201)323-2326, 2621

B.5.38 MIL-STD-2084

TITLE: General Requirements for Maintainability of Avionic and Electronic Systems and Equipment

DATE: 06 Apr 1982 - Notice 14 Jun 1983 **SUPERSEDES**:

SUMMARY: This standard supports the concepts of maintainability by design, considering modularization into assemblies suitable for test and replacement at the appropriate level maintenance for avionic and electronic systems and equipment. Maintenance terms are defined, as are the methods of analysis used to support computed maintenance parameters.

INDEX TERMS: Maintenance, Avionic Systems

OPR:
NAVAL AIR SYSTEMS COMMAND
Commanding Officer, Naval Air Engineering Center
Systems Engineering and Standardization Department (SESD) Code 53
ATTN: Mr. C. Meade
Lakehurst, NJ 08733-5100
(201)323-2326, 2621

B.5.39 MIL-STD-2107

TITLE: Product Assurance Program Requirements for Contractors

DATE: 08 Jul 1986 **SUPERSEDES**:

SUMMARY: This standard describes the requirements for a product assurance program applicable to contractors who provide products to the Navy. It provides contractor management with requirements for establishing and maintaining an acceptable product assurance program during the design, development and production of the required products.

INDEX TERMS: Quality Assurance

OPR:
NAVAL SEA SYSTEMS COMMAND (ORDINANCE SYSTEMS)
Commanding Officer, Naval Ordinance Station
Standardization Branch (Code 3730)
Indian Head, MD 20640-5000
(301)743-4250, 4358, 4510 4723

B.5.40 DOD-STD-2167A

TITLE: Defense System Software Development

DATE: 29 Feb 1988 **SUPERSEDES**: DOD-STD-2167 which superseded
DOD-STD-1679A (Navy) 22 Oct 1983;
MIL-STD-1644B (TD) 2 Mar 1984;
DOD-STD-2167, 4 Jun 1985

SUMMARY: This standard contains requirements for the development of Mission-Critical Computer System software. It establishes a uniform software development process which is applicable throughout the system life cycle. The software development process defines development activities which result in (1) the generation of different types and levels of software and documentation, (2) the application of development tools, approaches, and methods, and (3) project planning and control. It incorporates practices which have been demonstrated to be cost-effective from a life cycle perspective, based on information gathered by the DoD and industry.

Application to various types of software: this standard applies to deliverable software designated as Computer Software Configuration Items (CSCIs). This standard, or portions thereof, such as configuration management, quality evaluation, and documentation also applies to:

a. Software developed as part of a system or a Hardware Configuration Item but not explicitly identified as CSCI.

b. Non-deliverable software used in the development and testing of deliverable software and hardware (such as design and test tools).

c. Deliverable unmodified commercially available and reusable software.

d. Commercially available software, Government furnished software, and reusable software that is modified and delivered as part of the system.

The standard also provides standard formats for recording the information, such as plans, specifications, design, test, and code.

INDEX TERMS: System/Software Development, Life Cycle, Mission Critical Computer Resources (MCCR), Test and Evaluation

OPR:
SPACE AND NAVAL WARFARE SYSTEMS COMMAND
Commander Space and Naval Warfare Systems Command
SPAWAR-3212
Washington, DC 20363-5100
(703) 602-4493, 9188 (R. Singh)

B.5.41 DOD-STD-2168

TITLE: Defense System Software Quality Program
DATE: 29 Apr 1988 **SUPERSEDES**: MIL-S-52779A, 01 Aug 1979
SUMMARY: This standard contains requirements for the establishment and implementation of a software quality program. This program includes evaluation of the quality of software, associated documentation, and related activities, and the planning and follow-up activities necessary for timely and effective resolution of problems.

DOD-STD-2168 is intended to be used in conjunction with DOD-STD-2167A, Defense System Software Development, and with DOD-STD-7935A, DoD Automated Information Systems (AIS) Documentation Standard. These standards, together with other DoD and military specifications and standards governing configuration management, specification practices, project reviews and audits, and subcontractor control provide a means for achieving, determining, and maintaining quality in software and associated documentation.

The intent of this standard is to implement the policies of DoD Directive 4155.1, Quality Program, and to provide all of the necessary elements comprising a comprehensive quality program applicable to software development. This standard interprets the requirements of MIL-Q-9858, Quality Program Requirements, for software. In supersedes MIL-S-52779A, Software Quality Assurance Program Requirements. The software quality activities described in this standard are meant to be applied during each system life cycle phase in which software development or software change takes place. This standard incorporates the applicable requirements of MIL-STD-1520 and MIL-STD-1535.

INDEX TERMS: Software Quality Assurance, Software Development, Test and Evaluation
OPR: ARMAMENT MUNITIONS AND CHEMICAL COMMAND
Commander US Army Armament, Munitions and Chemical Command (AMCCOM)
US Army Armament Research, Development and Engineering Center
ATTN: SMCAR-ESC-S
Picatinny Arsenal, NJ 07806-5000
(201)724-6530, 6662, 6674

B.5.42 DOD-STD-5200.28

TITLE: Department of Defense Trusted Computer System Evaluation Criteria
DATE: December 1985 **SUPERSEDES**: 13 Aug 83
SUMMARY: This standard provides technical hardware/firmware/software security criteria and associated technical evaluation methodologies in support of the overall ADP system security policy, evaluation and approval/accreditation requirements for DoD AIS.
INDEX TERMS: Computer Security, Automated Information Systems (AIS)
OPR: National Computer Security Center

B.5.43 DOD-STD-7935A

TITLE: DoD Automated Information Systems (AIS) Documentation Standards
DATE: 31 October 1988 **SUPERSEDES**:
SUMMARY: This standard cover all types of technical documentation for AISs, applications computer programs, and revisions thereto; as well as the use of existing or developed standards for each document type. Excluded from the Scope is the Life Cycle Management Planning documentation and project request documentation.
INDEX TERMS: Automated Information Systems (AIS), Documentation
OPR: Office of Comptroller, B. Newlin, 697-9068

B.5.44 MIL-STD-DIF (DRAFT); PROJECT IPSC-0214

TITLE: Document Interchange Format (DIF)
DATE: 01 Mar 1985
SUMMARY: This document provides a standard set of codes which are needed and will be used by vendors who are installing office automation equipment in Navy activities. It defines the representation for control functions required by the Department of the Navy for text processing systems. It is important to stress that it is not the intent of this effort that manufacturers of text processing systems change their internal representation of these functions. Rather, manufactures will provide a common format for an agreed subset by a "filter" program, developed by the manufacturers, which will do the mapping of the DIF control functions from their internal representations to DIF representations on export and the reverse on import. DIF is intended for use within the existing framework of communications protocols. The utilization of DIF, therefore, would be to transmit the DIF data stream between devices via an appropriate communications methodology which permits transfer of an ASCII file in a transparent and reliable mode.
INDEX TERMS: Office Automation, Text Processing, Data Standards
OPR:

B.6 Military Handbooks

This section provides a synopsis of Military Handbooks related to software and systems.

B.6.1 MIL-HDBK-59

TITLE: Dept of Defense Computer Aided Acquisition and Logistic Support (CALS) Program Implementation Guide.
DATE: Oct 1989 (Draft)
SUMMARY: Provides guidance to program managers on how to apply CALS to weapons systems contracts and projects.
INDEX TERMS: Acquisition, Logistics, Program Management, Weapon Systems
OPR: OASD(P&L) Bill Gorham 756-8420

B.6.2 MIL-HDBK-245B

TITLE: Preparation of Statement of Work (SOW)
DATE: 01 June 1983, Notice 1 - 31 Dec 1987**SUPERSEDES**: MIL-HDBK-245A, 01 Aug 1978
SUMMARY: Provides general policy and detailed requirements for the writer of a Statement of Work to prepare a comprehensive statement. Sample outlines and do/don't guidance are provided.
INDEX TERMS: Acquisition, Contracting
OPR:
Commander, ARMAMENT MUNITIONS AND CHEMICAL COMMAND (AMCCOM)
US Army Research, Development and Engineering Center
ATTN: SMCAR-BAC-S
Picatinny Arsenal, NJ 07806-5000
(201) 724-6530, 6662, 6674

B.6.3 MIL-HDBK-286 (DRAFT)

TITLE: Software Quality Evaluation
DATE: 31 Dec 1985
SUMMARY: This handbook provides guidance to Development Agency and Software Support Agency personnel charged with planning, establishing, and managing a Software Quality Evaluation Program for a software development or support project. The handbook provides specific guidelines for applying DOD-STD-2168 to a software development or support project.
INDEX TERMS: Software Quality, Software Development
OPR:
ARMAMENT MUNITIONS AND CHEMICAL COMMAND
Commander US Army Armament, Munitions and Chemical Command (AMCCOM)
US Army Armament Research, Development and Engineering Center
ATTN: SMCAR-ESC-S
Picatinny Arsenal, NJ 07806-5000
(201)724-3531, Wayne Sherer

B.6.4 MIL-HDBK-287

TITLE: A Tailoring Guide for DOD-STD-2167A, Defense System Software Development
DATE: 11 Aug 1989
SUMMARY: This handbook aids in the tailoring of DOD-STD-2167A, Defense System Software Development. It contains overview material, specific DOD-STD-2167A tailoring considerations, tailoring methodology, and tailoring examples.
INDEX TERMS: System/Software Development, Documentation
OPR:
SPACE AND NAVAL WARFARE SYSTEMS COMMAND
Commander Space and Naval Warfare Systems Command (SPAWAR 3212)
Washington, DC 20363-5100
(202)602-9188, 4493, R. Singh

B.5.5 MIL-HDBK-334

TITLE: Evaluation of a Contractor's Software Quality Assurance Program
DATE: 15 Jul 1981
SUMMARY: This document provides basic and fundamental information and guidance to personnel concerned with the evaluation of a contractor's software quality assurance program, in connection with MIL-S-52779A.
INDEX TERMS: Quality Assurance, Contracting
OPR:
HQ USAISEC
ATTN: ASQB-TE
Ft. Huachuca, AZ 85613
(602) 538-7650

B.6.6 MIL-HDBK-782

TITLE: Software Support Environment Acquisition
DATE: 29 Feb 1988
SUMMARY: MIL-HDBK-782(AR) provides a common interpretation of the requirements and uses of DOD-STD-1467, Software Support Environment, and the information needed to use it effectively.
INDEX TERMS: Software Support Environment
OPR:
Commander, ARMAMENT MUNITIONS AND CHEMICAL COMMAND (AMCCOM)
US Army Research, Development and Engineering Center
ATTN: SMCAR-BAC-S
Picatinny Arsenal, NJ 07806-5000
(201) 724-6530, 6662, 6674

B.7 Military Specifications

This section provides a synopsis of Military Specifications related to software and systems.

B.7.1 MIL-D-28000 - IGES

TITLE: Digital Representation for Communications of Product Data: Initial Graphics Exchange Specification (IGES)
DATE: 20 Dec 1988 (Amendment 1)

SUMMARY: Provides the standard specification for interchange of Computer Aided Design / Computer Aided Manufacturing (CAD/CAM) information between DoD and industry computer systems.
INDEX TERMS: Standards, Interface Control
OPR: OASD(P&L) Don Hall 756-8420

B.7.2 MIL-M-28001 - SGML

TITLE: Markup Requirements and Generic Style Specification for Electronic Printed Output and Exchange of Text (Standardized General Markup Language)
DATE: 26 Feb 1988
SUMMARY: This standard specifies a method for exchanging revisable text files. It is designed to manage large publications which are subject to change, such as training manuals.
INDEX TERMS: Standards, Documentation
OPR: OASD(P&L) J. Dalgety 756-8420

B.7.3 MIL-R-28002 - Raster

TITLE: Raster Graphics Representation in Binary format, Requirements for
DATE: 30 Oct 1989 (Amendment 1)
SUMMARY: This standard specifies formats and procedures for exchanging large engineering drawings using bit mapped or Facsimile representations.
INDEX TERMS: Standards, Documentation
OPR: OASD(P&L) J. Dalgety 756-8420

B.7.4 MIL-D-28003 - CGM

TITLE: Digital Representation for communication of Illustration Data: Computer Graphics Metafile Application Profile
DATE: 20 Dec 88 FIPS Pub 28
SUMMARY: This standard specifies formats and procedures for exchanging of 2 dimensional vector illustrations, e.g. illustrations in a technical manual.
INDEX TERMS: Standards, Documentation
OPR: OASD(P&L) J. Dalgety 756-8420

B.7.5 MIL-H-46855B (Amendment 2)

TITLE: Human Engineering Requirements for Military Systems, Equipment, and Facilities
DATE: 05 Apr 1984 **SUPERSEDES:** MIL-H-46855B, 31 Jan 1979
SUMMARY: This specification defines the requirements for *applying human engineering* to the development of military systems. These include the analysis plans, design plans for equipment, facilities, and overall system definition including computer software. The human factors role throughout the development cycle is defined in terms of required plans, analyses, approvals, criteria, and testing. Actual design criteria are included by reference. A sample guide for application of this specification is included.
INDEX TERMS: Human Factors
OPR: US Army Missile Command
COMMANDER
ATTN: AMSMI-RD-SE-TD-ST
Redstone Arsenal, AL 35898-5270
(205)876-1335

B.7.6 MIL-Q-9858A (Amendment 2)

TITLE: Quality Program Requirements
DATE: 08 Mar 1985 **SUPERSEDES:** MIL-Q-9858A (Amendment 1), 07 Aug 1981
SUMMARY: An effective and economical quality program, planned and developed in consonance with the contractor's other administrative and technical programs, is required by this specification. Design of the program shall be based upon consideration of the technical and manufacturing aspects of production and related engineering design and materials. The program shall assure adequate quality throughout all areas of contract performance including design, development, fabrication, processing, assembly, inspection, test, maintenance, packaging, shipping, storage, and site installation.
INDEX TERMS: Quality Assurance
OPR: DIRECTORATE OF CONTRACTING AND MANUFACTURING POLICY, HQ USAF
Manufacturing and Quality Assurance Group
SAF/AQCM
Washington, DC 20330-1000
(202)695-4976

B.7.7 MIL-S-83490

TITLE: Specifications, Types, and Forms
DATE: 30 Oct 1968 **SUPERSEDES:**
SUMMARY: Prescribes general requirements for the preparation of specifications for the departments and agencies of DoD. It is a guide to the hierarchy of specifications procured under MIL-STD-490 (specification practices) military system procurement and MIL-STD-480 configuration control. It covers the level from the system specification, development specification, product specification, process specification, and material specification. It provides additional guidance on the form of specification from military standard to commercial practice.

INDEX TERMS: Specifications
OPR:

B.8 Army Regulations and Guidance

This section provides a synopsis of Army Regulations and Guidance related to software and systems.

B.8.1 AR 25-1

TITLE: *The Army Information Resource Management Program*
DATE: 18 Nov 1988 **SUPERSEDES**:
SUMMARY: This Army Regulation combines all policies relative to the Information Mission Area disciplines of automation, telecommunications, visual information, records management, publications, and printing and their related programs and library of management. The regulation realigns the responsibility for Information Resources Management in accordance with the reorganization of the Department of the Army Headquarters, executed in 1987. It describes the major programs that comprise and/or support the overall Army Information Resource Management Program.
INDEX TERMS: Automated Information Systems (AIS)
OPR: Headquarters
Department of the Army
Washington, D.C. 20310

B.8.2 AR 70-37

TITLE: Joint DoD Services/Agency Regulation, Configuration Management
DATE: 19 Jul 1976
SUMMARY: This Army Regulation provides integrated and uniform policies and guidance for configuration management across the armed forces and defense agencies.
INDEX TERMS: Configuration Management
OPR: Department of the Navy, Headquarters Naval Materiel Command

B.8.3 AR 70-XX

TITLE: Management of Army Mission Critical Computer Resources
DATE: 07 Nov 1985 **SUPERSEDES**:
SUMMARY: This regulation implements DoD Directive 5000.29. It establishes policy and assigns responsibilities for the life cycle management, planning budgeting, acquisition, testing, and support of computer resources employed by major and non-major mission critical Defense systems.
INDEX TERMS: Management, Mission Critical Computer Resources (MCCR)
OPR: Headquarters
Department of the Army
Washington, DC 20310

B.8.4 AR-70-1

TITLE: System Acquisition Policy and Procedures
DATE: 10 Oct 1988 **SUPERSEDES**:
SUMMARY: This Army regulation establishes responsibilities for the Army Acquisition Executive, Program Executive Officer (PEO), and Project/Product Manager (PM). It explores the DoD Life Cycle System Management Model as it applies to Army materiel acquisition. Chapter 9 specifically addresses the acquisition and management of Materiel Systems Computer Resources (MSCR). The scope of MSCR encompasses all computer hardware and software designed, configured, or acquired as an integral element of a materiel system that is required in order for the system to fully perform its mission. This chapter also identifies and defines activities of the Software Development Cycle.
INDEX TERMS: Acquisition, Computer Resources
OPR: Headquarters
Department of the Army
Washington, D.C. 20310

B.8.5 AR 1000-1

TITLE: Basic Policies for System Acquisition
DATE: 01 May 1981 **SUPERSEDES**:
SUMMARY: This regulation establishes basic Army policy and prescribes responsibilities and procedures for acquisition of materiel systems.
INDEX TERMS: Acquisition
OPR: Headquarters
Department of the Army (DAMA-RAA),
Washington, DC 20310-0635

B.8.6 DARCOM R-70-16

TITLE: Management of Computer Resources in Battlefield Automated Systems
DATE: 16 Jul 1979 **SUPERSEDES**:
SUMMARY: Implements DoD Directive 5000.29 Management of Computer Resources in Major Defense Systems. Establishes policy and assigns responsibilities for the planning, development, acquisition, testing, training and support of major and non-major Army battlefield automated systems employing computer resources. Addresses early computer resource planning, risk analysis, interoperability, training, quality, reuse, languages, and testing. States an intent to

insure that computer resources in Army battlefield automated systems are treated and managed as integral but subordinate parts of the overall system and not as separate or unique elements.
INDEX TERMS: Computer Resources, Mission Critical Computer Resources (MCCR), Battlefield Systems
OPR: Commander, US Army Materiel Command
ATTN: AMCDE-AT
5001 Eisenhower Ave.
Alexandria, VA 22333

B.8.7 CECOM-R 11-25

TITLE: Army Programs Ada Policy
DATE: 01 Jun 1988 **SUPERSEDES**:
SUMMARY: This regulation establishes the requirement for the use of the Ada programming language for all U.S. Army Communications-Electronics Command (CECOM) developed, managed, or supported Mission Critical Defense Systems (MCDS) and implements DoD Directives 3405.1 and 3405.2 within CECOM.
INDEX TERMS: Ada Programming Language, System/Software Development, Programming Languages, Weapon Systems
OPR: Headquarters, U.S. Army CECOM, Fort Monmouth, NJ 07703

B.8.8 CECOM-R 11-31

TITLE: Army Programs Computer Resource Management Policy
DATE: 23 Jan 1989 **SUPERSEDES**:
SUMMARY: This regulation establishes the CECOM Center for Software Engineering as the Computer Resource Management for all Mission Critical Defense Systems (MCDS) managed or supported by the U.S. Army Communications-Electronics Command (CECOM). Defines relationships and responsibilities of the development activity relative to MCDS managed or supported by CECOM. It defines minimum requirements for tasking by an activity external to CECOM relative to software engineering support to be accepted.
INDEX TERMS: Computer Resource
OPR: Headquarters, U.S. Army CECOM, Fort Monmouth, NJ 07703

B.8.9 CECOM-Policy:Compiler Distribution

TITLE: Policy for Providing Ada Compilers to Nonprofit Institutions of Higher Education
DATE: 29 Jul 1989 **SUPERSEDES**:
SUMMARY: This policy establishes a mechanism for providing Ada Compilers to Nonprofit Institutions of Higher Education that are under contract for basic or applied research.
INDEX TERMS: Compilers, Ada Programming Language, Education and Training
OPR: Headquarters, U.S. Army CECOM, Fort Monmouth, NJ 07703

B.8.10 CECOM-R (to be assigned)

TITLE: CECOM Software Acquisition and Support Policy for Mission Critical Defense Systems (MCDS)
DATE: 14 Jul 1989 **SUPERSEDES**:
SUMMARY: This regulation mandates the use on new CECOM contract solicitation the following procedures, standards and regulations:

a. CECOM-R 11-31, CECOM-R 11-25, DOD-STD-2167A, DOD-STD-2168, DOD-STD-1467, AMC 70-13 and AMC 70-14
b. The Software Engineering Institute (SEI) Software Capability Evaluation will be a requirement of the source selection process for all mission critical software procurements over $10 million.
c. Army Tactical Command and Control System (ATCCS) Common Hardware/Software, along with its associated Programming Support Environment (PSE), will be utilized to the maximum extent possible.
d. Software implemented in firmware, unless otherwise explicitly designated, will be treated as software and designated as a Computer Software Configuration Item.
e. Commercial, off-the-shelf Computer Aided Support Environment (CASE) tools will be used where possible and in preference to contractor developed environment tools.
f. All proposals to develop software for mission critical systems must address Total Quality Management.
g. Procurement packages requiring the acquisition or development of Weapon Systems and associated training devices software must have the concurrences of the Center for Software Engineering prior to contract award.
INDEX TERMS: Commercial Products, Acquisition, Mission Critical Computer Resources (MCCR),Quality Assurance, Software Support Environment
OPR: Headquarters, U.S. Army CECOM, Fort Monmouth, NJ 07703

B.9 Navy Instructions and Guidance

This section provides a synopsis of Navy Instructions and Guidance related to software and systems.

B.9.1 Secretary of the Navy Instructions

B.9.1.1 SECNAVINST 4130.2

TITLE: DON Configuration Management Policy
DATE: 11 MAY 87 **SUPERSEDES**: NAVMATINST 4130.1A, MCO 4130.1A,
NAVMATINST 4130.5, and NAVMATINST 4130.2A
SUMMARY: Establishes uniform requirements for the application and tailoring of Configuration Management for

material items acquired, operated or supported by DON. Provides a systematic means for documenting and controlling configuration of material items in order to better manage life cycle costs, performance, readiness and integrated logistic support.
INDEX TERMS: Configuration Management
OPR: Office of the Chief of Naval Operations (OP-43) Wash. DC 20350-1000

B.9.1.2 SECNAVINST 4200.32

TITLE: Design to Cost
DATE: 12 JUL 84 **SUPERSEDES**:
SUMMARY: Implements policies in DoD Directive 4245.3 regarding requirement to use "Design to Cost" principles for all ACAT I programs. Encourages use of principles therein for ACAT II programs and below as well.
INDEX TERMS: System/Software Design
OPR: SECNAV (ASN/CBM) Wash DC 20350-1000

B.9.1.3 SECNAVINST 4210.6A

TITLE: Acquisition Policy
DATE: 13 Apr 88 **SUPERSEDES**: SECNAVINST 4210.6
SUMMARY: Promulgates policy guidelines to improve and strengthen the acquisition process like: adherence to program initiation procedures, maximum use of competition and increased contractor investment.
INDEX TERMS: Acquisition
OPR: SECNAV (ASN/CBM) Wash DC 20350-1000

B.9.1.4 SECNAVINST 4210.7A

TITLE: Effective Acquisition of Navy Material
DATE: 16 JAN 87 **SUPERSEDES**: SECNAVINST 4210.7
SUMMARY: Establishes policies and assigns responsibilities to promote effective material acquisition through the use of non- development items to fulfill Navy requirements.
INDEX TERMS: Acquisition
OPR: Commander, Space and Naval Warfare Systems Command
Washington, DC 20362-5100

B.9.1.5 SECNAVINST 4210.9

TITLE: Acquisition and Management of Technical Data and Computer Software
DATE 25 Jan 88 **SUPERSEDES**: SECNAVNOTE 4210 of 30 Jan 87
SUMMARY: Establishes policies and assigns responsibilities in DDN for acquisition of a technical data package to define a product baseline for procurement and reprocurement of a production item, spare or repair parts.
INDEX TERMS: Acquisition
OPR: Secretary of the Navy (ASN S&L)
Washington, DC 20350-1000

B.9.1.6 SECNAVINST 4855.1

TITLE: Quality Assurance Program
DATE 10 Sep 89 **SUPERSEDES**:
SUMMARY:
INDEX TERMS: Quality Assurance
OPR: Secretary of the Navy
Washington, DC 20350-1000

B.9.1.7 SECNAVINST 4858.2E

TITLE: DON Value Engineering Program
DATE: 06 Jul 84 **SUPERSEDES**: SECNAVINST 4858.2D
SUMMARY: Establishes policy and procedures for implementing Value Engineering.
INDEX TERMS: Value Engineering
OPR: Secretary of the Navy
Washington, DC 20350-1000

B.9.1.8 SECNAVINST 5000.1C

TITLE: Major and Non-Major Acquisition Programs
DATE: 16 SEP 88 **SUPERSEDES**: SECNAVINST 5000.1B
SUMMARY: Establishes policies within DON to govern acquisition programs.
INDEX TERMS: Acquisition
OPR: Secretary of the Navy (ASN S&L)
Washington, DC 20350-1000

B.9.1.9 SECNAVINST 5000.2

TITLE: Major and Non-Major Acquisition Program Procedures
DATE: 1 NOV 88 **SUPERSEDES**: OPNAVINST 5000.42C
SUMMARY: Establishes implementation procedures for DON implementation of defense acquisition program procedures.
INDEX TERMS: Acquisition

OPR: Secretary of the Navy (ASN S&L)
Washington, DC 20350-1000

B.9.1.10 SECNAVINST 5000.39A

TITLE: Acquisition and Management of Integrated Logistic Support (ILS) for Systems and Equipment
DATE: 03 Mar 1986 **SUPERSEDES:** SECNAVINST 5000.39
SUMMARY: Implements DoD Directive 5000.39; establishes policies and assigns responsibilities for ILS.
INDEX TERMS: Acquisition, Logistics, Management
OPR: OASN(S&L) (AP)

B.9.1.11 SECNAVINST 5200.18

TITLE: Data Elements and Data Codes Standardization Program
DATE: 03 Dec 68 **SUPERSEDES:** SECNAVINST 10462.11A
SUMMARY: Implementation of Data Elements and Data Codes Standardization Program within DON.
INDEX TERMS: Data Elements, Standards
OPR: Commander, Naval Data Automation Command
Washington, DC 20374-1662

B.9.1.12 SECNAVINST 5200.19

TITLE: Data Elements and Data Codes Standardization Procedures
DATE: 09 Dec 69 **SUPERSEDES:** OMINST 10462.1
SUMMARY: Implements Data Elements and Data Codes Standardization Procedures within DON.
INDEX TERMS: Data Elements, Standards
OPR: Commander, Naval Data Automation Command
Washington, DC 20374-1662

B.9.1.13 SECNAVINST 5200.24

TITLE: Implementation of Standard Data Elements and Related Features
DATE: 03 Nov 69 **SUPERSEDES:**
SUMMARY: Implements DON Standard Data Elements and related features.
INDEX TERMS: Data Elements, Standards
OPR: Commander, Naval Data Automation Command
Washington, DC 20374-1662

B.9.1.14 SECNAVINST 5200.32

TITLE: Management of ECR in Department of the Navy Systems
DATE: 11 JUN 79 **SUPERSEDES:** N/A
SUMMARY: Implements DoD Directives 5000.29 and 5000.31 within DON and supplements policies and procedures for management of Navy weapons, communications, command and control and intelligence systems when embedded computer resources are incorporated as integral components.
INDEX TERMS: Management, Embedded Computer Resources (ECR)
OPR: Department of the Navy Information Resources Management
Washington, DC 20350-1000

B.9.1.15 SECNAVINST 5200.37

TITLE: Acquisition of Software-Intensive C^2 Information Systems
DATE: 5 JAN 88 **SUPERSEDES:** N/A
SUMMARY: Defines acquisition policy for software-intensive command and control information systems. Promotes routine user involvement in software development process; encourages rapid fielding of needed capabilities.
INDEX TERMS: Acquisition, Command, Control, Communications and Intelligence (C^3I)
OPR: Secretary of the Navy (ASN RE&S)
Washington, DC 20350-1000

B.9.1.16 SECNAVINST 5230.3B

TITLE: Information Technology User Group Program
DATE: 14 NOV 86 **SUPERSEDES:** SECNAVINST 5230.3A
SUMMARY: Implements DoD Instruction 7930.1 as charter for the Information Technology User Group. Intended to foster cooperation, coordination and communication among DoD components in the area of utility and systems software.
INDEX TERMS: Manpower and Personnel
OPR: Commander, Naval Data Automation Command
Washington, DC 20374-1662

B.9.1.17 SECNAVINST 5230.4

TITLE: DON ADP Program
DATE: 03 May 76 **SUPERSEDES:** SECNAVINST 5200.25
SUMMARY: Establishes DON ADP Program
INDEX TERMS: Automated Information Systems (AIS)
OPR: Department of the Navy Information Resources Management
Washington, DC 20350-1000

B.9.1.18 SECNAVINST 5230.8

TITLE: Information Processing Standards for Computers Program
DATE: 10 May 82 **SUPERSEDES**: SECNAVINST 5200.28
SUMMARY: Issues policies and procedures governing the IPSC program.
INDEX TERMS: Standards
OPR: Commander, Naval Data Automation Command
Washington, DC 20374-1662

B.9.1.19 SECNAVINST 5230.9A

TITLE: Information Resources Program Planning
DATE 16 Oct 85 **SUPERSEDES**: SECNAVINST 5230.9
SUMMARY: Reviews policies and responsibilities for program planning.
INDEX TERMS: Management
OPR: Commander, Naval Data Automation Command
Washington, DC 20374-1662

B.9.1.20 SECNAVINST 5230.10

TITLE: DON Strategic Plan for Managing Information and Related Resources
DATE: 1 APR 87 **SUPERSEDES**:
SUMMARY: Establishes consolidated guidance for managing information, information systems, computer resources, other information technologies and related resources supporting all DON missions. Establishes long-range planning guidance for each of these functional management areas.
INDEX TERMS: Management, Computer Resources
OPR: Department of the Navy Information Resources Management
Washington, DC 20350-1000

B.9.1.21 SECNAVINST 5231.1B

TITLE: Life Cycle Management Policy and Approval Requirements for Information System Projects
DATE: 8 MAR 85 **SUPERSEDES**: Various: See Instruction
SUMMARY: Provides a standard discipline for managing information systems projects by changing emphasis in DON from managing various types of information technology to managing information systems. Adapts the system acquisition process for information system projects.
INDEX TERMS: Acquisition, Management
OPR: Department of the Navy Information Resources Management
Washington, DC 20350-1000

B.9.1.22 SECNAVINST 5231.3A

TITLE: Information System Executive Board (ISEB)
DATE: 25 APR 89 **SUPERSEDES**: SECNAVINST 5231.3
SUMMARY: Establishes DON ISEB Review Council for information system projects requiring approval at the DON level or above.
INDEX TERMS: Acquisition, Management
OPR: Department of the Navy Information Resources Management
Washington, DC 20350-1000

B.9.1.23 SECNAVINST 5232.1

TITLE: Quality Assurance Program for Information System Projects
DATE 16 Oct 85
SUPERSEDES: SECNAVINST 10462.18 and OPNAVINST 10462.14
SUMMARY: Establishes Quality Assurance program for Information Systems projects in DON.
INDEX TERMS: Quality Assurance
OPR: Commander, Naval Data Automation Command
Washington, DC 20374-1662

B.9.1.24 SECNAVINST 5233.1B

TITLE: DoD Automated Data Systems Documentation Standards
DATE: 25 JAN 79 **SUPERSEDES**: SECNAVINST 5233.1A
SUMMARY: DON implementation of the DoD standard and instructions for the preparation of documents.
INDEX TERMS: Documentation
OPR: Commander, Naval Data Automation Command
Washington, DC 20374-1662

B.9.1.25 SECNAVINST 5234.1B

TITLE: Federal Compiler Testing Center
DATE 06 Nov 80 **SUPERSEDES**: SECNAVINST 5234.1A
SUMMARY: Implements DON use of the Federal Compiler Testing Center and establishes policy for COBOL compiler validation.
INDEX TERMS: Programming Languages
OPR: Commander, Naval Data Automation Command
Washington, DC 20374-1662

B.9.1.26 SECNAVINST 5234.2

TITLE: Computer Programming Language Policy
DATE: 3 NOV 88 **SUPERSEDES**: N/A
SUMMARY: Implements DoD Directive 3405.1 and DoD Directive 3405.2. Specifies that software be acquired based on analysis of life-cycle costs and impact, using the following order of preference: (1) non-developmental item (NDI) software, (2) Ada-based software tools, and (3) approved standard HOLs. Mandates use of Ada for mission-critical systems (with some exceptions) and required Ada for all other applications unless use of another approved HOL (listed in an enclosure) is more cost effective over the life cycle of the application. To achieve the long range goal of transition to Ada, requires Ada compilers be validated, encourages use of Ada-based program design language and modern software engineering principles. Establishes responsibilities and waiver process.
INDEX TERMS: Ada Programming Language, Programming Language
OPR: Department of the Navy Information Resources Management
Washington, DC 20350-1000

B.9.1.27 SECNAVINST 5236.1B

TITLE: Contracting for ADP Resources
DATE 15 Oct 80 **SUPERSEDES**: SECNAVINST 5236.1A
SUMMARY: Revises policies relating to contracting for ADP resources in DON.
INDEX TERMS: Acquisition, Computer Resources, Contracting
OPR: Commander, Naval Data Automation Command
Washington, DC 20374-1662

B.9.1.28 SECNAVINST 5236.2A

TITLE: ADP Services Contracts
DATE 07 Jul 80 **SUPERSEDES**: SECNAVINST 5236.2
SUMMARY: Promulgates policies and procedures governing the acquisition of ADP services.
INDEX TERMS: Acquisition, Computer Resources, Contracting
OPR: Commander, Naval Data Automation Command
Washington, DC 20374-1662

B.9.1.29 SECNAVINST 5236.4

TITLE: ADP Software Exchange and Release
DATE 17 Feb 88 **SUPERSEDES**: SECNAVNOTE 5236 of 19 May 86
SUMMARY: Implements policy and procedures for the exchange and release of DON ADP software.
INDEX TERMS: Automated Information Systems (AIS), Software Exchange
OPR: Commander, Naval Data Automation Command
Washington, DC 20374-1662

B.9.1.30 SECNAVINST 5237.2

TITLE: Economical Use of Timesharing Services
DATE 17 Feb 81 **SUPERSEDES**:
SUMMARY: Provides procedure for economical use of commercial computer timesharing services attained from teleprocessing service program (TSP)
INDEX TERMS: Computer Resources
OPR: Commander, Naval Data Automation Command
Washington, DC 20374-1662

B.9.1.31 SECNAVINST 5238.1C

TITLE: Computer Resources Management
DATE 07 Apr 89
SUPERSEDES: SECNAVINST 5237.1, SECNAVINST 5237.3, and SECNAVINST 5238.1B
SUMMARY: Establishes policies and procedures for inventory management, sharing, and reutilization or redistribution of computer resources within DON.
INDEX TERMS: Computer Resources, Equipment Redistribution
OPR: Department of the Navy Information Resources Management
Washington, DC 20350-1000

B.9.1.32 SECNAVINST 5239.2

TITLE: Department of the Navy Automated Information Systems
DATE 15 Nov 89 **SUPERSEDES**:
SUMMARY: Establishes the Department of Navy AIS security program, sets forth policies and guidelines for the program, and defines organizational responsibilities for executing the program elements.
INDEX TERMS: System/Software Security, Automated Information Systems
OPR: Department of the Navy Information Resources Management
Washington, DC 20350-1000

B.9.1.33 SECNAVINST 5420.176

TITLE: Interlaboratory Computing
DATE 08 Oct 74 **SUPERSEDES**:
SUMMARY: Establishes Navy Laboratory Computing Committee.

INDEX TERMS: Computer Resources, Research and Development
OPR: Secretary of the NAVY (SO-3)
Washington, DC 20350-1000

B.9.1.34 SECNAVINST 5420.188B

TITLE: Navy and Marine Corps Program Decision Meetings
DATE: 17 JAN 89
SUPERSEDES: SECNAVINST 5420.188A and 4210.8A
SUMMARY: Provides guidance regarding streamlining of the acquisition review process and provides procedures for conducting Navy and Marine Corps Program Decision Meetings.
INDEX TERMS: Acquisition, Management
OPR: Secretary of the Navy (ASN RE&S)
Washington, DC 20350-1000

B.9.1.35 SECNAVINST 5430.98

TITLE: DONIRM, Reorganization of
SUPERSEDES: SECNAVINST 5224.1
SUMMARY: Creates the Office of the Department of the Navy Information Resources Management (DONIRM) with sole responsibility within DON for IRM.
INDEX TERMS: Computing Resources, Management
OPR: Department of the Navy Information Resources Management
Washington, DC 20350-1000

B.9.1.36 SECNAVINST 5230.11

TITLE: Information Resources Management (IRM) Review Program
DATE 19 Oct 89 **SUPERSEDES**: SECNAVINST 10462.18
SUMMARY: Establishes DONIRM Review Program
INDEX TERMS: Computer Resources, Management
OPR: Department of the Navy Information Resources Management
Washington, DC 20350-1000

B.9.1.37 SECNAVINST 10550.4

TITLE: Policies for the Use of Microelectronics in Navy Systems and Equipment
DATE 01 Nov 67 **SUPERSEDES**:
SUMMARY: Disseminates DoD policy for use of microelectronics in military systems and equipment.
INDEX TERMS: Microelectronics, Computer Hardware
OPR: Commander, Space and Naval Warfare Systems Command
Washington, DC 20362-5100

B.9.2 OPNAV Instructions

B.9.2.1 OPNAVINST 3960.10C

TITLE: Test and Evaluation
DATE: 14 SEP 87 **SUPERSEDES**: OPNAVINST 3960.10B
SUMMARY: Defines test and evaluation (T&E) responsibilities. Establishes procedures for planning, conducting and reporting T&E. Delineates the relationships of various phases. Established procedures and formats for Test and Evaluation Master Plans (TEMPs). Establishes procedures for obtaining RDT&E support for R&D that is not part of an acquisition program. Specifically addresses the T&E of Mission Critical Computer Resources. Addresses the T&E of block upgrades to software, the software annex to the TEMP and the T&E of Navy Standard Embedded Computer Resources.
INDEX TERMS: Test and Evaluation (T&E), Test and Evaluation Master Plan (TEMP), Mission Critical Computer Resources (MCCR)
OPR: Chief of Naval Operations (OP-983)
Washington, DC 20350-2000

B.9.2.2 OPNAVINST 4790.2B

TITLE: The Naval Aviation Maintenance Program
DATE: 26 MAY 82 **SUPERSEDES**: OPNAVINST 4790.2A
SUMMARY: This instruction is the basic document and authority governing management of all Naval Aviation Maintenance. Affects handling of Fleet hardware which is part of the weapon system, and supporting items. Maintenance procedures for equipment used in the operating system or its support are updated. Command, administrative and management relationships are established, with policies and procedures for the assignment of maintenance tasks and/or responsibilities for conduct of the Program.
INDEX TERMS: Aviation System, Weapon System, Maintenance
OPR: Chief of Naval Operations (OP-514)
Washington, DC 20350-2000

B.9.2.3 OPNAVINST 5000.42C (Revision in draft)

TITLE: Research, Development and Acquisition Procedures
DATE: 10 MAY 86 **SUPERSEDES**: OPNAVINST 5000.42B of 13 APR 85

SUMMARY: Amplifies general guidance in DoD Directive 5000.1. This instruction has largely been superseded by SECNAVINST 5000.2. Program initiation portions remain in effect and are being refined in an upcoming revision.
INDEX TERMS: Acquisition, Research and Development, Management
OPR: Chief of Naval Operations (OP-98)
Washington, DC 20350-2000

B.9.2.4 OPNAVINST 5200.28

TITLE: Life Cycle Management of Mission Critical Computer Resources (MCCR) for Navy Systems Managed Under the Research, Development and Acquisition Process
DATE: 24 Sep 86 **SUPERSEDES**: N/A
SUMMARY: Establishes policy for the acquisition, management and life-cycle support of software and related computer resources in amplification of SECNAV and OPNAV policies. Also incorporates the provisions of the Standard Embedded Computer Resources Review Program (SECR Review Council).
INDEX TERMS: Acquisition, Life Cycle Management, Embedded Computer Resources (ECR)
OPR: Department of the Navy Information Resources Management
Washington, DC 20350-1000

B.9.2.5 OPNAVINST 5239.1A

TITLE: Department of the Navy Automatic Data Processing Security Program
DATE: 3 AUG 82 **SUPERSEDES**: OPNAVINST 5239.1
SUMMARY: Establishes DON ADP Security Program for all ADP activities and networks and assigns responsibilities. Executive briefs on ADP Security and DON Security Manual are contained in enclosures.
INDEX TERMS: System/Software Security
OPR: Department of the Navy Information Resources Management
Washington, DC 20350-1000

B.9.3 Naval Materiel Command (NAVMAT) Instructions, Guidance and TADSTANDS

B.9.3.1 NAVMATINST 3960.6B

TITLE: Test and Evaluation
DATE: 06 Feb 1981 **SUPERSEDES**:
SUMMARY: Identifies test and evaluation planning requirements for ACAT I, II, and III programs, and guidelines for ACAT IV programs. Identifies and reviews procedures leading to certification of readiness for Operational Evaluation. Test and Evaluation Master Plan review for ACAT I and II programs in the areas of planning, logistics, reliability and maintainability, and other disciplines are prescribed.
INDEX TERMS: Test and Evaluation (T&E), Test and Evaluation Master Plan (TEMP)
OPR: SPACE AND NAVAL WARFARE SYSTEMS COMMAND
Commander, Space and Naval Warfare Systems Command
Washington, DC 20363-5100

B.9.3.2 NAVMATINST 4130.1A, AR 70-37

TITLE: Joint DoD Services/Agency Regulation, Configuration Management
DATE: **SUPERSEDES**:
SUMMARY: Provides integrated and uniform policies and guidance for configuration management across the armed forces and defense agencies. Of general application to all development conducted by the DoD. Detail level for software is minimal.
INDEX TERMS: Configuration Management
OPR: SPACE AND NAVAL WARFARE SYSTEMS COMMAND
Commander, Space and Naval Warfare Systems Command
Washington, DC 20363-5100

B.9.3.3 NAVMATINST 4130.2A

TITLE: Configuration Management of Computer Software Associated with Tactical Data Systems and Other Technical Computer Systems Developed by or for the Naval Materiel Command
DATE: 19 Jul 1976 **SUPERSEDES**:
SUMMARY: Provides policy and procedures for configuration management to be applied to the acquisition and maintenance support of software for tactical digital and technical computer systems under Naval Materiel Command management. Directs that policy and procedures be carried out. Directs creation of a Software Change Control Board in most instances. Outlines the responsibilities of the Board and the use of Interface Design Specifications for software development and life cycle maintenance.
INDEX TERMS: Configuration Management, Life Cycle Management
OPR: SPACE AND NAVAL WARFARE SYSTEMS COMMAND
Commander, Space and Naval Warfare Systems Command
Washington, DC 20363-5100

B.9.3.4 NAVMATINST 5200.27A, MAT 09Y:RSF

TITLE: Transfer of Navy Tactical Digital System Software Responsibility; Procedures for
DATE: 18 Apr 1973 **SUPERSEDES**:
SUMMARY: Provides procedures for the transfer of Navy tactical digital software responsibility form a development activity to a program maintenance activity. Directs that procedures be followed. Two enclosures provide operating

procedures for the transfer of responsibility and an example of a suitable milestone chart format for planning such transfers. Defines and directs the planning documents and procedures that the development activity must prepare or follow. Describes the liaison between the development and program maintenance activity during software development. Describes the responsibilities of the development activity for support contractor efforts.
INDEX TERMS: Configuration Management, Life Cycle Management, Maintenance
OPR: SPACE AND NAVAL WARFARE SYSTEMS COMMAND
Commander, Space and Naval Warfare Systems Command
Washington, DC 20363-5100

B.9.3.5 NAVSO P-3656

TITLE: Department of the Navy Handbook for Implementation of Non-Developmental Item Acquisition
DATE: Jun 1988 **SUPERSEDES**:
SUMMARY: This handbook is a guide for acquisition managers and functional personnel who are, or will be, involved in Non-Developmental Item (NDI) acquisition. A definition of NDI and basic NDI concepts are presented. Feasibility investigation and analysis, solicitations and source selection, product assurance, test and evaluation and logistic support of NDI in the acquisition process are presented.
INDEX TERMS: Acquisition, Non-Developmental Item (NDI)
OPR: SPACE AND NAVAL WARFARE SYSTEMS COMMAND
Commander, Space and Naval Warfare Systems Command
Washington, DC 20363-5100

B.9.3.6 TADSTAND A, 08Y/DCR Ser 230 T-9

TITLE: Standard Definitions for Embedded Computer Resources in Tactical Digital Systems
DATE: 02 Jul 1980 **SUPERSEDES**:
SUMMARY: Establishes standard definitions for terms relating to embedded computer resources (ECR) in tactical digital systems.
INDEX TERMS: Embedded Computer Resources (ECR), Standards
OPR: SPACE AND NAVAL WARFARE SYSTEMS COMMAND
Commander, Space and Naval Warfare Systems Command
SPAWAR-3212
Washington, DC 20363-5100
(703) 602-4493, 9188 (M. Romeo, R. Singh, A. Selgas)

B.9.3.7 TADSTAND B (Rev 2), MAT 08Y

TITLE: Standard Embedded Computers, Displays, Peripherals, and Input/Output Interfaces
DATE: 02 Jan 1985 **SUPERSEDES**:
SUMMARY: Establishes both planned and current standard Navy embedded computers, computer peripherals, and I/O interfaces. Describes conditions under which these standards may be waived and the procedures to follow to obtain such a waiver.
INDEX TERMS: Embedded Computer Resources (ECR), Interface Control
OPR: SPACE AND NAVAL WARFARE SYSTEMS COMMAND
Commander, Space and Naval Warfare Systems Command
SPAWAR-3212
Washington, DC 20363-5100
(703) 602-4493, 9188 (M. Romeo, R. Singh, A. Selgas)

B.9.3.8 TADSTAND C (Rev 2), SPAWAR Ser 321/2063

TITLE: Computer Programming Language Standardization Policy for Tactical Digital Systems
DATE: 28 Jan 1988 **SUPERSEDES**: TADSTAND C (Rev 1)
SUMMARY: Promulgates policy for the standardization of computer programming languages used in the development, acquisition, deployment, and support of tactical digital systems. Use of low-level code is restricted by this TADSTAND. Defines both approved and planned approved programming languages. Mandates the use of Ada and Ada-based program design languages in writing software for mission-critical systems. Identifies the conditions under which the CMS-2 Languages may be used. Defines approved programming language preprocessors. Ada-based test languages are preferred, but C/ATLAS may be used as the first-choice alternative to Ada. Describes the procedures required to request waivers.
INDEX TERMS: Ada Programming Language, CMS-2 Programming Language
OPR: SPACE AND NAVAL WARFARE SYSTEMS COMMAND
Commander, Space and Naval Warfare Systems Command
SPAWAR-3212
Washington, DC 20363-5100
(703) 602-4493, 9188 (M. Romeo, R. Singh, A. Selgas)

B.9.3.9 TADSTAND D, SPAWAR, Ser 321/2129

TITLE: Reserve Capacity Requirements for Mission-Critical Systems
DATE: 27 Oct 1989 **SUPERSEDES**:
SUMMARY: Establishes hardware reserve capacity requirements for unplanned, unknown future growth. Requires a 50% reserve capacity for main memory, secondary storage, processor throughput, number of input/output channels, and input/output channel throughput. Describes the procedure required to request waivers.

INDEX TERMS: Computer Hardware, Reserve Capacity
OPR: SPACE AND NAVAL WARFARE SYSTEMS COMMAND
Commander, Space and Naval Warfare Systems Command
SPAWAR-3212
Washington, DC 20363-5100
(703) 602-4493, 9188 (M. Romeo, R. Singh, A. Selgas)

B.9.3.10 TADSTAND E (Rev 2)

TITLE: Software Development, Documentation, and Testing Policy for Navy Mission Critical Systems
DATE: 24 Jan 1989 **SUPERSEDES**: TADSTANDs 2, 3, and 9
SUMMARY: Promulgates standardization policy for the development, documentation,, and testing of software used in the development, acquisition, deployment, and support of Navy mission critical systems. Mandates that all software for applicable mission critical systems be developed, documented, tested, and supported in accordance with DOD-STD-2167A and DOD-STD-2168. Mandates that all software for applicable mission-critical systems be developed, documented, tested, and supported in accordance with DOD-STD-2167A, DOD-STD-2168, and associated Data Item Descriptions. The standards are also required when more than 30% of existing software is changed. Requires tailoring of the standards to improve quality and reduce cost. Prescribes Navy-unique acceptance testing and stress testing requirements. Identifies program manager's activities when the standards are used.
INDEX TERMS: System/Software Development, Documentation, Mission Critical Computer Resources (MCCR), Test and Evaluation (T&E)
OPR: SPACE AND NAVAL WARFARE SYSTEMS COMMAND
Commander, Space and Naval Warfare Systems Command
SPAWAR-3212
Washington, DC 20363-5100
(703) 602-4493, 9188 (M. Romeo, R. Singh, A. Selgas)

B.9.4 NAVAIR Instructions and Guidance

B.9.4.1 NAVAIRINST 3960.2A

TITLE: Test and Evaluation
DATE: 11 Aug 1978 **SUPERSEDES**:
SUMMARY: NAVAIR supporting instruction for OPNAVINST 3960.10 for implementation and to assign responsibilities within NAVAIR as a function of Project scope. The Advanced Development Project Officer, Project Manager, and/or Acquisition Manager are defined and identified with overall project responsibility; AIR-05 and AIR-06 are designated as sharing the functional responsibility for T&E. Details are provided as to the role in the review and approval of test plans by the NAVAIR structure.
INDEX TERMS: Test and Evaluation (T&E), Program Management
OPR:

B.9.4.2 NAVAIRINST 4000.14A

TITLE: Navy-Prepared Integrated Logistic Support Plans and Operational Logistic Support Plans for Aeronautical Systems and Equipment
DATE: 03 June 1975 **SUPERSEDES**:
SUMMARY: Promulgates the procedures and requirements for the development and preparation of the logistic support plans and equipment to be procured by or for the Naval Air Systems Command HQ. The enclosures provide instructions and formats for preparation or distribution of Integrated Logistic Support (ILS) Requirements Outline, ILS Plans, and Operational Logistic Support Plans.
INDEX TERMS: Integrated Logistics Support (ILS), Aeronautical Systems
OPR:

B.9.4.3 NAVAIRINST 4130.1B

TITLE: NAVAIR Configuration Management Manual
DATE: **SUPERSEDES**:
SUMMARY: Comprehensive background in the Configuration Management process, including requirements for procedures to evaluate, implement, and record configuration changes for NAVAIR hardware and software. Contains flow charts to illustrate the processes and sample forms as guides. Covers life cycle of hardware configuration items from concept definition through full-scale development and the operational life of the system. Software configuration management procedures are similarly covered from the Software Configuration Management Plan in concept development, through the final product baseline and test reports. Defines the Software Configuration Control process through system development and initial deployment. Configuration Management during the post-deployment software support phase is addressed in NAVAIRINST 5230.6
INDEX TERMS: Configuration Management
OPR:

B.9.4.4 NAVAIRINST 4200.14B

TITLE: Policy and Guidelines for Procurement of Data and Specific Acquisition of Unlimited Rights in Technical Data
DATE: May 1979 **SUPERSEDES**:
SUMMARY: Establishes policy and provides procedures for the acquisition of unlimited rights in technical data by NAVAIR and Field Agencies. Covers hardware, software, and process data. This instruction is intended for use with

Defense Acquisition Regulations (DAR) and MIL-STD-1679. DAR 9-601 providing definitions, and DAR 7-104.9(a) dealing with the use of "limited rights in software" markings should be consulted.
INDEX TERMS: Acquisition, Contracting, Data Rights
OPR:

B.9.4.5 NAVAIRINST 4275.3E

TITLE: Implementation of Configuration Control for DOD-STD-480A and MIL-STD-481A
DATE: 08 Dec 1982 **SUPERSEDES**:
SUMMARY: Provides specific guidance for selecting and implementing the Configuration Control Requirements of DOD-STD-480A and MIL-STD-481A. Defines the difference between the two on the basis of responsibility for an entire system (DOD-STD-481A). Specific actions which are required by the inclusion of either standard within a contract structure are outlined in detail. Affects configuration management procedures and scope of change analysis at either component or system level.
INDEX TERMS: Configuration Management
OPR:

B.9.4.6 NAVAIRINST 5100.3B

TITLE: Naval Air Systems Command System Safety Program
DATE: **SUPERSEDES**:
SUMMARY:
INDEX TERMS: System/Software Safety, Aeronautical Systems
OPR:

B.9.4.7 NAVAIRINST 5215.8C, AIR 4105P

TITLE: The NAVAIR Technical Directive System
DATE: 02 Mar 1979 **SUPERSEDES**:
SUMMARY: Describes the policy and procedures governing the NAVAIR HQ Technical Directive (TD) System. The TD System is the authorized medium for directing the accomplishment and recording of modifications and one-time inspections of NAVAIR accepted equipment. The software component of fielded NAVAIR systems may be the subject of a TD bulletin. Changes to the software in fielded NAVAIR systems will be documented by a TD change.
INDEX TERMS: Engineering Changes, Management
OPR:

B.9.4.8 NAVAIRINST 5230.5

(canceled; included in NAVAIRINST 5230.11)
TITLE: Responsibility and Requirements for Preparation of Software Life Cycle Management Plans (SLCMP)
DATE: 21 Jul 1976 **SUPERSEDES**:
SUMMARY: Identifies activities responsible for the preparation and maintenance of SLCMPs. Enclosure provides detailed instructions for the format and content of SLCMPs. The SLCMP addresses the operational software requirements for the complete life cycle of the weapon system. It is mandatory that the requirements described in the SLCMP be included in any Request for Proposals issued for full-scale development. Upon approval by the Project or Acquisition Manager, the SLCMP shall become the governing document for operational software life cycle support.
INDEX TERMS: Life Cycle Management, Acquisition, Management, Mission Critical Computer Resources (MCCR)
OPR:

B.9.4.9 NAVAIRINST 5230.6

(under revision)
TITLE: Establishment of Mission-Critical Computer Resources Software Change Review Boards (SCRB)
DATE: 14 June 1983 **SUPERSEDES**:
SUMMARY: Provides for the establishment of SCRBs. Enclosures provide a SCRB charter and a list of SCRB guidance documents. Directs project or acquisition managers to establish SCRBs during the validation or full-scale development phases when the project or acquisition manager judges such a formal review structure to be in NAVAIR's interests. Directs the creation of an SCRB prior to Fleet introduction unless all affected organizations and activities agree it is not necessary.
INDEX TERMS: Mission Critical Computer Resources (MCCR), Acquisition, Engineering Changes, Test and Evaluation (T&E)
OPR:

B.9.4.10 NAVAIRINST 5230.9

(under revision)
TITLE: Policy and Procedures for the Establishment and Operation of Naval Air Systems Command Systems Software Support Activities
DATE: 14 Jun 1983 **SUPERSEDES**:
SUMMARY: Establishes the requirements for software support activities and promulgates policy, procedures, responsibilities and operating relationships pertaining to their mission, functions, directions, and support. Specifically designates Navy activities to be assigned software support by warfare/functional category.
INDEX TERMS: Management, Software Support Activities
OPR:

B.9.4.11 NAVAIRINST 5230.11 (DRAFT)

TITLE: Policy and Procedures for the Development of Mission-Critical Computer Resources (MCCR) as part of Naval Air Systems Command Weapon System/Equipment
DATE: **SUPERSEDES**:
SUMMARY: Provides policy, procedures, and assigns responsibility for the development of Mission-Critical Computer Resources (MCCR) as part of weapon system/equipment procurements. Establishes requirements for Computer Resources Working Groups (CRWG) and Computer Resources Life-Cycle Management Plans (CRLCMP). The CRLCMP replaces the Software Life-Cycle Management Plan (SLCMP) previously required by NAVAIRINST 5230.5).
INDEX TERMS: Mission Critical Computer Resources (MCCR), Life Cycle Management, Acquisition, Management, Weapon Systems
OPR:

B.9.4.12 NAVAIRINST 5400.14C

TITLE: The Cognizant Field Activity Program
DATE: 27 Dec 1982 **SUPERSEDES**:
SUMMARY: This establishes and explains the procedures by which items of NAVAIR equipment being developed are put under the cognizant of designated field activities. It explains the responsibilities of these agencies and NAVAIR in the procurement of Service-approved equipment through its life cycle of support. Various items of equipment comprising a complete weapons system must be supported through the appropriate cognizant agency. This affects the maintenance of government furnished equipment used in facilities as part of the mission system. In addition, any proposed changes in configuration for such equipment must be properly coordinated.
INDEX TERMS: Weapon Systems, Management
OPR:

B.9.4.13 AV-2000A

TITLE: Format for Naval Air Systems Command Avionic Equipment Performance Specifications
DATE: 14 Nov 1983 **SUPERSEDES**:
SUMMARY: This document provides the format of the performance specification for equipment. Each paragraph explains its use and available options. The format includes information on the marking of software documentation, defines software media (tape, disks), and defines applicable interface data which may include bus structures and formats. In addition, material is provided relative to the programming of microprocessors and for the use of HOC in the design.
INDEX TERMS: Avionic Systems, Specifications
OPR:

B.9.4.14 AV-10000A, Supplement 1

TITLE: Examples of paragraphs for Specifying Avionics System Performance
DATE: 02 June 1982 **SUPERSEDES**:
SUMMARY: This document supplements the AV-1000 format by providing specific examples of paragraph wording, tables, and figures. It contains several examples of not only general information on computer software, but specifics on support programs and operational programs. The functional description of these capabilities in performance terms is shown.
INDEX TERMS: Avionic Systems, Specifications, Software Documentation
OPR:

B.9.4.15 AV-10000B

TITLE: Format for Naval Air Systems Command Avionic System Performance Specifications for Weapons Systems
DATE: 02 Jun 1983 **SUPERSEDES**:
SUMMARY: Contains the basic specification format for an overall weapon system specification, based upon MIL-STD-961 and MIL-STD-490, together with examples of the use of the format for specific systems. Extensive information on Avionic Systems (computer) software documentation and a computer program abstract outline is provided.
INDEX TERMS: Avionic Systems, Specifications, Weapon Systems, Documentation
OPR:

B.9.5 SPAWAR Instructions

B.9.5.1 SPAWARINST 5200.22

TITLE: Naval Electronic Systems Command Computer Resources Acquisition Management
DATE: 28 Feb 1979 **SUPERSEDES**:
SUMMARY: This instruction established basic policy of the Naval Electronic Systems Command with regard to computer resources acquisition management, placing particular emphasis on the development of embedded computer software products.
INDEX TERMS: Embedded Computer Resources (ECR), Mission Critical Computer Resources (MCCR), Acquisition, Management
OPR:

B.9.5.2 SPAWARINST 5200.23 (Ch-1)

TITLE: SPAWAR Computer Software Life-Cycle Management Guide
DATE: 01 Mar 1979 **SUPERSEDES**:
SUMMARY: The purpose of the Computer Software Life Cycle Management Guide is to provide information to the

SPAWAR program manager to allow him to understand and be able to meet sound software development practices for a SPAWAR software system acquisition.
INDEX TERMS: Life Cycle Management, Program Management, System/Software Development
OPR:

B.9.6 Chief, Naval Education & Training (CNET) Instructions

B.9.6.1 CNET Instruction (CNETINST) 1500.21

TITLE: Development of Interactive Courseware in Support of Instructional Systems
DATE: **SUPERSEDES**:
SUMMARY:
INDEX TERMS: *Education and Training, Computer Aided Instruction*
OPR:

B.9.7 NAVSEA Instructions

B.9.7.1 NAVSEAINST 4130.12

TITLE: LCM for Ships, Systems, Equipment and Computer Programs
DATE: **SUPERSEDES**:
SUMMARY:
INDEX TERMS: Life Cycle Management
OPR:

B.9.7.2 NAVSEAINST 4855.25

TITLE: Computer Software Product Quality Program
DATE: **SUPERSEDES**:
SUMMARY:
INDEX TERMS: Quality Assurance
OPR:

B.9.7.3 NAVSEAINST 5450.41B

TITLE: Missions and Functions of FCDSSA's
DATE: **SUPERSEDES**:
SUMMARY:
INDEX TERMS: Maintenance
OPR:

B.9.7.4 NAVSEAINST 5450.49

TITLE: Life Cycle Support Agent (LCSA) for Navy Standard Support Software
DATE: 9 Nov 1984 **SUPERSEDES**:
SUMMARY: This NAVSEA Instruction assigns responsibility for being the NAVSEA Standard Support Software Life Cycle Support Agent (LCSA) to FCDSSA, San Diego. This instruction also describes the correction and change request processes to be followed by users of CMS-2L, CMS-2M, CMS-2Y support software.
INDEX TERMS: Software Support Activities
OPR: Commander, Naval Sea Systems Command, PMS-412

B.9.8 Marine Corps Orders

B.9.8.1 MCO 3093.1C

TITLE: Intraoperability and Interoperability of Marine Corps Tactical C^4I Systems
DATE: 15 June 1989 **SUPERSEDES**: MCO 3093.1B, 5 June 1987
SUMMARY: Implements interoperability policies directed by the Secretary of Defense/Joint Chiefs of Staff. Establishes policies and management procedures within the Marine Corps necessary to ensure that both Marine Corps intraoperability and joint/combined interoperability standards are implemented in Marine Corps tactical command and control, communications, computer and intelligence (C^4I) systems.
INDEX TERMS: Interoperability, Command, Control, Communications, and Intelligence (C^3I)
OPR: HQMC (C^2I), Maj Nick Hoffer, 694-4522

B.9.8.2 MCO P3900.XX (Draft)

TITLE: Systems Engineering Manual
DATE: 19 May 1989**SUPERSEDES**: MCO 3080.1, 4700.3, 4120.11
SUMMARY: Publishes guidance and procedures to implement the Systems Engineering portions of MCO P5000.10, DoD Directive 5000.3 and DoD Directive 5000.4 in the Marine Corps.
INDEX TERMS: Systems Engineering, Quality Assurance
OPR: MCRDAC(PSE-P), Doug Heinz, 694-5540

B.9.8.3 Marine Corps Bulletin 3900

TITLE: Tactical Software Management and Funding
DATE: 07 January 1988 **SUPERSEDES**: N/A
SUMMARY: Publishes new policy decisions regarding the management and funding of software support for tactical systems.
INDEX TERMS: Management, Funding, Weapon Systems

OPR: HQMC (C^4I^2/C^2IC). Capt. Doug Smith, AV 224-1589

B.9.8.4 MCO P4105.XX (Draft)

TITLE: Integrated Logistics Support (ILS) Manual
DATE: **SUPERSEDES**:
SUMMARY: Further expands and defines the procedures for acquisition and management of Integrated Logistic Support (ILS) prescribed by MCO P5000.10.
INDEX TERMS: Integrated Logistics Systems (ILS)
OPR: MCRDAC (PSL-P). Jim Riordan. AV 227-2603

B.9.8.5 MCO 4130.2

TITLE: Configuration Management
DATE: 17 Oct 1977
SUMMARY: This Marine Corps Order prescribes policies and procedures, and assigns responsibilities for configuration management support of computer programs and equipment of designated field tactical systems and their training support systems.
INDEX TERMS: Configuration Management
OPR:

B.9.8.6 MCO P4130.8

TITLE: Configuration Management Manual
DATE: 04 January 1989 **SUPERSEDES**: MCO 4130.1 through 4130.7
SUMMARY: Provides policy and procedures required to implement Configuration Management within the Marine Corps, and to provide a systematic methodology for documenting and controlling the configuration of material and equipment.
INDEX TERMS: Configuration Management
OPR: MCRDAC (PSE)

B.9.8.7 MCO P5000.10C

TITLE: Systems Acquisition Management Manual
DATE: 01 April 1989 **SUPERSEDES**: MCO 5000.10B
SUMMARY: Publishes management guidance and procedures which implement the policies in MCO 5000.15 for the acquisition of weapon systems, computer resources, and equipment within the Marine Corps.
INDEX TERMS: Acquisition
OPR: MCRDAC (SASST2)

B.9.8.8 MCO 5000.15

TITLE: Marine Corps Systems Acquisition Management Policy
DATE: 19 February 1985 **SUPERSEDES**: MCO 3900.3D
SUMMARY: Establishes policy and assigns general command and staff responsibilities for systems acquisition management in the Marine Corps.
INDEX TERMS: Acquisition
OPR: HQMC (RDD-29)

B.9.8.9 MCO 5200.23A

TITLE: Management of Mission-Critical Computer Resources (MCCR) in the Marine Corps
DATE: 30 December 1986 **SUPERSEDES**: MCO 5200.23, 19 August 1982
SUMMARY: Implements DoD Directives 3405.1&2, 5000.29, SECNAVINST 5200.32, & 5234.2. Establishes policy for the acquisition, management, and life-cycle support of MCCR in the Marine Corps.
INDEX TERMS: Mission-Critical Computer Resources (MCCR), Computer Resources, Acquisition, Management
OPR: HQMC (C^4I^2/C^2IC), Capt. Doug Smith, AV 224-8726

B.9.8.10 MCO 5230.2D

TITLE: Designation of Marine Corps Central Design and Programming Activities (MCCDPAs)
DATE: 11 April 1983 **SUPERSEDES**:
SUMMARY: Establishes three MCCDPAs within the Marine Corps that provide technical support in the design, programming, testing, implementation, distribution, documentation, and maintenance of Marine Corps Class I software.
INDEX TERMS: System/Software Development
OPR: HQMC (CCI)

B.9.8.11 MCO 5230.15

TITLE: Data Base Administration
DATE: 09 August 1983 **SUPERSEDES**:
SUMMARY: Establishes the organization and responsibilities for data administration as it applies to the design, development, implementation, and management of data bases within the Marine Corps.
INDEX TERMS: Data Base Administration
OPR: HQMC (CCI)

B.9.8.12 MCO P5231.1

TITLE: Life Cycle Management for Information Systems Projects
DATE: 17 September 1987 **SUPERSEDES**:

SUMMARY: Establishes Marine Corps policy and regulations governing the development, implementation, operation, and management of information systems projects.
INDEX TERMS: Life Cycle Management
OPR: HQMC (CCI)

B.9.8.13 MCO 5234.4

TITLE: Configuration Management for ADP System Software
DATE: 31 May 1984**SUPERSEDES**:
SUMMARY: Establishes standard procedures for planning, justifying, testing, evaluating, acquiring, and implementing ADP systems software in the Marine Corps.
INDEX TERMS: Configuration Management
OPR: HQMC (CCI)

B.9.8.14 MCO 5271.1

TITLE: Information Resources Management (IRM) Standards and Guidelines Program
DATE: 19 September 1986 **SUPERSEDES**:
SUMMARY: Establishes the IRM Standards and Guidelines Program, and guides the implementation of Marine Corps IRM policy by the development and distribution of publications that set technical standards for the management of all IRM activities.
INDEX TERMS: Standards
OPR: HQMC (CCI)

B.9.8.15 MCO 5271.2

TITLE: Information Resources Management (IRM) Program Planning
DATE: 29 September 1986 **SUPERSEDES**:
SUMMARY: Establishes policy and objectives and assigns responsibilities for IRM Program planning.
INDEX TERMS: Management, Computer Resources
OPR: HQMC (CCI)

B.9.8.16 MCO 5271.3

TITLE: Management Oversight of Information Systems
DATE: 16 March 1988 **SUPERSEDES**:
SUMMARY: Assigns responsibilities for management oversight of information systems and supporting information resources. The assigned responsibilities address systems development, system operation, and the use of hardware and system software in support of developing and operational information systems.
INDEX TERMS: Management, Computer Resources
OPR: HQMC (CCI)

B.9.8.17 MCO P5510.14

TITLE: ADP Security Manual
DATE: 02 January 1981 **SUPERSEDES**:
SUMMARY: Establishes policy and procedures for ADP security by addressing physical security, communications, emanations, hardware, software, procedural, risk management, contingency planning and other security aspects contributing to the protection of an ADP system, site, facility, or operation.
INDEX TERMS: ADP, Software Security
OPR: HQMC (CCI)

B.9.8.18 MCO 5311.5 (Draft)

TITLE: Marine Corps Software Support Personnel Model (SSPRM)
DATE: 18 August 1989 **SUPERSEDES**: N/A
SUMMARY: Establishes the SSPRM as the official Marine Corps estimating tool for mission-critical software support personnel requirements, and provides policy on the use, maintenance, and re-validation of the SSPRM.
INDEX TERMS: Cost Estimating, Management
OPR: HQMC (C^4I^2/C^2IC), Capt. Dough Smith, 694-8726

B.10 Air Force Regulations and Guidance

B.10.1 AFR 57-4

TITLE: Operational Requirements, Modification Approval and Management
DATE: 28 August 1987 **SUPERSEDES**: AFR 57-4, 23 May 1983
SUMMARY: Modification programs offer the Air Force ways to correct deficiencies in or improve the capabilities of existing Air Force equipment and nonnuclear munitions in lieu of new development programs. This publication states the procedures for planning, documenting, obtaining approval, and managing the modification. It applies to the processing of modification requirements for all Air Force, Air Reserve Forces, and Security Assistance activities for which the Air Force has logistic support responsibility. It partially implements DoD Directive 5000.1 and Instruction 500.2. It should be used along with AFR 57-1, Operational Needs, and configuration control portions of AFR 65-3, Configuration Management, that pertain to modifications.
INDEX TERMS: Management, Maintenance, Re-Engineering
OPR: AF/LEYYS

B.10.2 AFR 122-9

TITLE: Nuclear Surety Design Certification Program for Nuclear Weapon System Software and Firmware
DATE: 24 Aug 1987
SUMMARY: This regulation establishes the requirements for nuclear surety design certification of nuclear weapon system software and firmware and defines the management process for the software and firmware design certification effort. It also identifies requirements for management of design-certified software and firmware components and defines the certification effort that can be used in obtaining certification. This regulation applies to all units with a mission involving the development of software and firmware for use in nuclear weapon systems or in equipment that interfaces with these weapon systems. It does not apply to US Air Force Reserve and Air National Guard units and members.
INDEX TERMS: Software Certification, Nuclear Systems, Weapon Systems, System/Software Safety, Reliability
OPR: AFISC/SNA

B.10.3 AFR 700-4, Vol I

TITLE: Communications-Computer Systems Program Management and Acquisition, Communication-Computer Systems Program Management
DATE: 15 March 1985 **SUPERSEDES**: AFR 100-18, 15 Feb 83; AFR 100-19, 15 Feb 83; and Chapters 3, 7, 11, and 12 of AFR 300-6, 11 Jul 80; Sections A, B, and C of Chapter 3 and Chapters 4 and 7, Volume I, AFR 300-12, 12 Sep 77; and AFR 300-15, 16 Jan 78.
SUMMARY: This regulation provides policy to activities requiring, implementing, supporting, or acquiring communications-computer systems capabilities as defined in AFR 700-1, Managing Air Force Communications-Computer Systems. This regulation is one of the communications-computer systems publications prescribed by AFR 700-12, Developing and Processing Communications-Computer Systems Publications, and applies to all organizations in the Air Force, Air Reserve Forces, and other agencies involved in acquiring communications-computer systems capabilities through Air Force channels.
INDEX TERMS: Communications, Management, Acquisition.
OPR: AF/SCTX

B.10.4 AFR 700-9, Vol I

TITLE: Information Systems Standards, Information Systems Standardization Program
DATE: 15 March 1985 **SUPERSEDES**: Sections 1 and 2, Air Force Pamphlet 300-1, 18 Sep 84; paragraphs 1, 2, 3, 4, 5, 11, 12, 13, and 14, AFR 300-4, Vol I, 11 May 83; AFR 300-5, 27 May 80; AFR 300-10, 15 Dec 76; and AFR 300-16, 28 Jun 79.
SUMMARY: This regulation establishes the Information Systems Standardization Program and sets policies and procedures for applying and developing information systems standards. It applies to all Air Force activities which identify requirements for, procure, develop, maintain, document, or use Air Force information systems, as defined in AFR 700-1, Managing Air Force Information Systems. It implements applicable parts of DoD Instruction 4120.20, 28 Dec 76; DoD Directive 4120.3, 10 Feb 79; DoD Directive 4120.21, 13 Nov 80; DoD Directive 5000.11, 7 Dec 64 and DoD Instruction 5000.31, 24 Nov 76. This is one of the information systems publications prescribed by AFR 700-12, Developing and Processing Information Systems Publications.
INDEX TERMS: Standards
OPR: AF/SCTX

B.10.5 AFR 700-26

TITLE: Management of Small Computers
DATE: 15 December 1988 **SUPERSEDES**: AFR 700-26, 30 Apr 86
SUMMARY: This regulation provides policy and guidance for planning, acquiring, managing, and operating small computers and related software. It applies to all Air Force activities including Air Force Reserve and Air National Guard units and members. This regulation implements DoD-7950.1-M, Defense Automation Resources Management Manual, DoD Directive 7920.1, Life Cycle Management of Automated Information Systems, and Federal Information Resources Management Regulations (FIRMR) and Federal Acquisition Regulations (FAR), as appropriate.
INDEX TERMS: Small Computers, Management, Acquisition
OPR: AF/SCTX

B.10.6 AFR 700-53

TITLE: Management of Standard Systems
DATE: 01 May 1989**SUPERSEDES**: N/A
SUMMARY: This regulation establishes policy, procedure, and responsibility for the operation, maintenance, acquisition, and control of standard communications-computer systems and components. For brevity, "standard communications-computer systems" will be referred to as "standard systems" throughout this regulation. It specifies agencies and individuals responsible for establishing and implementing standard system policy. It applies to all Air Force personnel, including Air Force Reserve and Air National Guard units and members. The term "major command" (MAJCOM), as used in this regulation, includes separate operating agencies and direct reporting units.
INDEX TERMS: Communications, Management
OPR: AF/SCTX

B.10.7 AFATL Pamphlet 800-1

TITLE: Air Force Armament Laboratory Mission Critical Computer Resources Purchase Request Package Preparation Guide

DATE: 15 Oct 87 **SUPERSEDES**:
SUMMARY: This pamphlet provides information and guidance for AFATL program managers to use in preparing Purchase Request packages involving mission critical computer resources. In short, this guide provides guidance on tailoring DOD-STD-2167A deliverables for R&D software acquisitions
INDEX TERMS: Management, Computer Resources, Acquisition, Mission Critical Computer Resources (MCCR)
OPR: AFATL/FXG

B.10.8 AFR 800-2

TITLE: Acquisition Program Management
DATE: September 1985 **SUPERSEDES**: AFR 800-2, 13 Aug 82
SUMMARY: This regulation, with AFRs 57-1 and 55-24, prescribes the system acquisition process for programs funded primarily through procurement appropriations; the Security Assistance Program; or the Research, Development, Test and Evaluation (RDT&E) appropriation. These regulations apply from initiation (identification of the mission need) through concept exploration, demonstration and validation, full-scale development, production, and deployment phases. While the acquisition strategy is unique for each system, those basic policies and principles described here are the same. This regulation applies to all Air Force programs and to joint programs for which the Air Force is designated the lead service. It implements applicable sections of DoD Directive 5000.1, 12 March 1986; DoD Instruction 5000.2, 12 March 1986; and DoD Directive 7920.1, 17 October 1978. All persons involved in acquisition programs, including major modifications, must comply with this regulation.
INDEX TERMS: Acquisition, Management
OPR: SAF/AQXA

B.10.9 Air Force Operational Test and Evaluation Center Pamphlet 800.2

TITLE: Air Force Operational Test and Evaluation Center Pamphlet: Software Operational Test and Evaluation Guidelines
DATE: 01 Aug 1986 **SUPERSEDES**:
SUMMARY: This pamphlet is a guide for the Air Force Operational Test and Evaluation Center Software Evaluation Manager and Deputy for Software Evaluation. It describes the numerous activities associated with planning, conducting, analyzing and reporting software operational test and evaluation assessments. Volumes cover Management of Software Operational Test and Evaluation, Software Maintainability, Software Operator-machine Interface and Software Maturity.
INDEX TERMS: Test and Evaluation (T&E)
OPR: Air Force Operational Test and Evaluation Center

B.10.10 AFR 800-3

TITLE: Engineering for Defense Systems
DATE: June 1977 **SUPERSEDES**: AFR 800-3, 1 Jun 76
SUMMARY: This regulation outlines policy for the management of a totally integrated engineering effort, under the program management concept established in AFR 800-2. It also outlines the engineering effort that is typically applied, phase by phase, throughout the acquisition life cycle. It applies to each Air Force organization engaged in, or supporting, acquisition programs as defined in AFR 800-2, and implements DoD Directives 4140.43, 5 Dec 75 and C-4600.3, 21 May 76.
INDEX TERMS: System/Software Engineering, Acquisition
OPR: SAF/AQXM

B.10.11 AFSC/AFLC Pamphlet 800-5

TITLE: Software Independent Verification and Validation (IV&V)
DATE: 20 May 88 **SUPERSEDES**:
SUMMARY: The purpose of this pamphlet is to help program directors develop an IV&V program that meets their system's specific requirements. The pamphlet describes a six-step procedure for determining the need for a software IV&V effort, establishing its scope, identifying tasks and subtasks associated with each IV&V requirement, selecting a qualified agent, and estimating software IV&V costs. In addition, this pamphlet integrates the software engineering tasks of DOD-STD-2167A with software IV&V tasks to make sure value is added to the software development process and product. The methods used in this pamphlet are based on a MIL-STD-882 (System Safety Program Requirements) approach, as well as a composite of similar initiatives from Space Division, Aeronautical Systems Division, and Electronic Systems Division. It applies to all AFSC/AFLC activities involved with software acquisition management. It does not apply to Air National Guard or US Air Force Reserve units and members.
INDEX TERMS: Management, Computer Resources, Test and Evaluation (T&E), Independent Verification and Validation (IV&V)
OPR: HQ AFSC/PLR and HQ AFLC/MMT

B.10.12 Aeronautical Systems Division Pamphlet 800-5

TITLE: Software Development Capability/Capacity Review
DATE: 10 Sep 87 **SUPERSEDES**:
SUMMARY: This pamphlet provides guidance for planning and conducting the Software Development Capability/Capacity Review (SDCCR) as an integral part of the system/subsystem acquisition source selection process. The SDCCR is intended to assess each of the bidder's ability to develop the software required by the program Request for Proposal (RFP). The SDCCR was defined to apply to Full Scale Development (FSD) programs. This review could also be applied to Demonstration and Validation Programs if the software generated during this phase is envisioned to be carried into the FSD program. This review covers the total software development process, including management,

technical, and personnel resources. This pamphlet is based on the experiences gained in reviewing software developments on defense system programs over the past 20 years and on conducting the SDCCR on Aeronautical Systems Division programs starting in 1963. The guidance provided should be helpful as a baseline for planning, organizing and conducting future SDCCRs. This review has been expanded to cover Ada technology.
INDEX TERMS: Management, System/Software Development
OPR: Aeronautical Systems Division/EN/CRFP

B.10.13 AFR 800-14

TITLE: Management of Computer Resources
DATE: Sep 1986 **SUPERSEDES**: AFR 800-14 Vol. I, 12 Sep 1976
SUMMARY: This Air Force Regulation consolidates policy and procedures pertaining to the acquisition, development, and support of computer resources acquired under the AFR-800 series of regulations. It applies to all activities responsible for planning, developing, acquiring, supporting and using computer resources in systems acquired and managed under the AFR 800-2 program management concept.
INDEX TERMS: Management, Computer Resources, Acquisition, System/Software Development
OPR: AF/AQX

B.10.14 AFSC Pamphlet 800-14

TITLE: Software Quality Indicators
DATE: 20 Jan 87 **SUPERSEDES**:
SUMMARY: This pamphlet describes indicators that will provide insight into the quality of mission-critical computer resources. It is intended to help program managers by presenting indicators that reflect the quality of the software products developed in an acquisition program. It also provides information that reflects experience gained on previous acquisition programs. Indicators are just that: indicators. They do not, nor are intended to, replace sound quality practices. These indicators, properly applied and meticulously followed-up, will lead the contractor and program office to those areas requiring additional quality attention. This pamphlet does not apply to the Air National Guard or US Air Force Reserve units and members.
INDEX TERMS: Management, Acquisition, System/Software Development, Metrics
OPR: HQ AFSC/PLR

B.10.15 AFSC/AFLC Supplement 1 - AFR 800-14

TITLE: Lifecycle Management of Computer Resources in Systems
DATE: 14 Sep 87 **SUPERSEDES**:
SUMMARY: Defines computer resources acquisition management and support responsibilities to specific AFSC and AFLC organizations.
INDEX TERMS: Acquisition, Management, Computer Resources
OPR: HQ AFSC/PLR and HQ AFLC/MMT

B.10.16 AFSC Pamphlet 800-43

TITLE: Software Management Indicators
DATE: 31 Jan 86 **SUPERSEDES**:
SUMMARY: This pamphlet describes management indicators that will provide visibility into the acquisition of mission-critical computer resources. It is intended to help program managers by presenting software management indicators that reflect the status of software development in an acquisition program. It also provides information that reflects experience on previous acquisition projects. Indicators are just that: indicators. They do not, nor are they intended to, replace sound management practices and communications. Indicators, properly applied, thoroughly understood, and meticulously followed-up, will lead the contractor and program office to those areas requiring management attention. This pamphlet does not apply to Air National Guard or US Air Force Reserve units and members.
INDEX TERMS: Management, Metrics, Acquisition
OPR: HQ AFSC/PLR

B.10.17 AFSC/AFLC Pamphlet 800-45

TITLE: Software Risk Abatement
DATE: 30 Sep 88 **SUPERSEDES**:
SUMMARY: This pamphlet describes software risk abatement processes, composed of risk identification, analysis, and handling techniques, that can significantly contribute to improving the acquisition of mission-critical computer resources. This pamphlet is intended to help program directors by integrating software risk abatement with system-level risks-handling techniques. Risk abatement techniques can help the contractor and program office improve the performance and support of the software in weapon systems. This pamphlet applies to all AFSC and AFLC activities involved and software acquisition management. It does not apply to the Air National Guard or to the US Air Force Reserve units and members.
INDEX TERMS: Management, Computer Resources, Risk Reduction
OPR: HQ AFSC/PLR and HQ AFLC/MMT

B.10.18 AFSC Pamphlet 800-51 (DRAFT)

TITLE: Software Development Capability Assessment
DATE: Oct 89 **SUPERSEDES**:
SUMMARY: This pamphlet describes a set of methods to assess and evaluate a prospective contractor's software development capability. It provides guidance in the preparation, execution, analysis, and reporting of the results of a

software development capability assessment. While the information presented here is not all-inclusive, it should be used as the basis for developing a contractor specific software engineering capability assessment and analyzing the results. This method, when properly applied, will lead to an understanding of the contractor's capability to meet the software development costs, quality, and schedule. This pamphlet does not apply to the Air National Guard or the US Air Force Reserve Units and members.
INDEX TERMS: Management, Metrics, System/Software Development
OPR: HQ AFSC/PLR

B.11 SDIO Directives

B.11.1 SDS Software Policy

TITLE: Strategic Defense System Software Policy
DATE: 25 October 1989 **SUPERSEDES**:
SUMMARY: The Strategic Defense Initiative Organization (SDIO) is responsible for exploring and demonstrating key technologies associated with the concepts of defense against nuclear weapon bearing ballistic missiles. Software is a keystone technology for the Phase I Strategic Defense System (SDS). SDIO views the current problems associated with the development of quality military software to be both real and urgent and is therefore adopting a strategy to promote a change of attitudes, policies, and practices concerning software acquisition.

The SDIO has reviewed the Report of the Defense Science Board Task Force on Military Software [DSB87] and in general concurs with the Task Force's conclusions and recommendations. SDIO is therefore adopting the Defense Science Board Task Force Report as the cornerstone of its Software Policy.

The SDIO recognizes that, in its role as a leader of advanced technology development within the Department of Defense, it should create and foster an acquisition and management environment which encourages, promotes and rewards the use of modern software engineering practices. To this end, the SDS Software Policy provides guidance in the usage of appropriate software engineering practices for the development of all mission-critical, full-scale development SDS software.
INDEX TERMS: Languages, Ada, Prototyping, Supportability, Risk Reduction, System/Software Security, Configuration Management, Software Support Environments, Documentation, Portability, Test and Evaluation (T&E), Reuse, Data Rights
OPR:

B.11.2 SDIO Directive 3405

TITLE: Software Policy
DATE: 25 October 1989 **SUPERSEDES**:
SUMMARY: SDIO implementation of the SDS Software Policy.
INDEX TERMS: Languages, Ada, Prototyping, Supportability, Risk Reduction, System/Software Security, Configuration Management, Software Support Environments, Documentation, Portability, Test and Evaluation (T&E), Reuse, Data Rights
OPR:

B.12 Defense Communications Agency (DCA) Instructions and Guidance

B.12.0.1 DCA Memorandum, H102

TITLE: Interim Policy for Use of Ada Programming Language
DATE: 31 August 1989 **SUPERSEDES**:
SUMMARY: The Memorandum is written in accordance with the authority contained in DoD Directive 3405.1, Computer Programming Language Policy, April 1987. It provides policy guidance for the use of the Ada programming language in software systems over 1000 lines of code.
INDEX TERMS: Ada, Programming Language
OPR: Willie Garrett 202/692-0998

B.12.0.2 DCA Instruction 630-125-1

TITLE: Data Elements & Data Codes Standardization and Procedures
DATE: (under revision) **SUPERSEDES**: 20 August 1965 version
SUMMARY: Implements DoD Directive 5000.11. This Instruction implements the DoD policies and procedures concerning the DoD Data Elements and Data Codes program and assigns responsibilities for accomplishment of program objectives within the DCA.
INDEX TERMS: Data Elements, Data Codes, Standards
OPR: Willie Garrett 202/692-0998

B.12.0.3 DCA Instruction 630-125-2

TITLE: Implementation of the Standard Data Elements and Related Features Program
DATE: (under revision) **SUPERSEDES**: 10 June 1975 version
SUMMARY: Implements DoD Directive 5000.18. This Instruction establishes the policy and procedures for implementation of DoD standard data elements and related features in DCA data systems.
INDEX TERMS: Data Elements, Data Codes, Standards
OPR: Willie Garrett 202/692-0998

B.12.0.4 DCA Instruction 630-230-9

TITLE: Transferability of Computer Programs
DATE: (under revision) **SUPERSEDES**: 18 Dec. 1967 version
SUMMARY: DCA unique instruction. This Instruction provides policy concerning the preparation of computer programs to ensure a high level of transferability of software among computers of different types.
INDEX TERMS: Programming Language, Portability
OPR: Willie Garrett 202/692-0998

B.12.0.5 DCA Instruction 630-230-17

TITLE: DoD Automated Data System Documentation Standards
DATE: 4 May 1983 **SUPERSEDES**: 9 Dec. 1977 version
SUMMARY: Implements DOD-STD 7935A. This Instruction provides guidelines for the development and revision of the documentation for an automated information system (AIS) or applications software, and specifies the content of each of the 11 types of documents that may be produced during the life cycle of an AIS.
INDEX TERMS: Documentation, Automated Information Systems (AIS)
OPR: Willie Garrett 202/692-0998

B.12.0.6 DCA Instruction 630-230-19

TITLE: Security Requirements for Automatic Data Processing (ADP) Systems
DATE: (under revision) **SUPERSEDES**: 29 Oct. 1985 version
SUMMARY: Implements DoD Directive 5200.28. This Instruction prescribes policy, assigns responsibilities, and provides procedures for the DCA ADP Security Program for ADP activities and local area networks.
INDEX TERMS: System/Software Security, Automated Data Processing (ADP)
OPR: Willie Garrett 202/692-0998

B.12.0.7 DCA Instruction 630-230-21

TITLE: Life Cycle Management of Automated Information Systems (AIS)
DATE: (under revision) **SUPERSEDES**: 30 June 1980 version
SUMMARY: Implements DoD Directive 7920.1. This Instruction establishes policy and procedures and delineates responsibility for review, approval, and life cycle management of major automated information systems.
INDEX TERMS: Automated Information Systems (AIS)
OPR: Willie Garrett 202/692-0998

B.12.0.8 DCA Instruction 630-230-23

TITLE: ADP Software Exchange and Release
DATE: 7 Aug. 1980 **SUPERSEDES**:
SUMMARY: Implements DoD Instruction 7930.2. This Instruction establishes the policy and delineates responsibility for the exchange and release of ADP software.
INDEX TERMS: Automated Data Processing (ADP), Exchange and Release
OPR: Willie Garrett 202/692-0998

B.12.0.9 DCA Instruction 630-230-28

TITLE: Baselining of Automated Information Systems (AIS)
DATE: 17 July 1987 **SUPERSEDES**:
SUMMARY: Implements DoD Instruction 7920.4. This Instruction is published in accordance with the authority contained in DoD Instruction 7920.4, Baselining of Automated Information Systems (AIS), 21 March 88. It establishes the policy and procedures and delineates responsibility for baselining AIS programs.
INDEX TERMS: Baselining, Automated Information Systems (AIS)
OPR: Willie Garrett 202/692-0998

B.13 National Security Agency Policy and Guidance

B.13.1 NSAM 81-2

TITLE: NSA/CSS Software Acquisition Manual
DATE: 15 May 1986 **SUPERSEDES**: NSAM 81-2, 21 Dec 1978
SUMMARY: Establishes policies on the acquisition and development of software systems for the National Security Agency/Central Security Service.
INDEX TERMS: Acquisition, Software Development
OPR: NSA

B.13.2 NSAM 81-3/MIL-STD-1703 (NS)

TITLE: NSA/CSS Software Product Standards Manual
DATE: 15 April 1987 **SUPERSEDES**: NSAM 81-3, 26 Jul 1979
SUMMARY: This standard describes a set of documents and activities that comply with the policies of the NSA/CSS software Acquisition Manual (NSAM 81-2). It contains formats and guidance for the following documents: Software Development Plan, Software Standards and Practices Manual, Software Quality Assurance Plan, Software Configuration Management Plan, Software Requirements Specification, Software System/Subsystem Specification, Interface Control Document, Software Program Specification, Data Dictionary Document, Unit Development Folders, General Unit Test Plan, Software System Integration Test Plan, Software System Developmental Test Plan, Software Test Procedures, Software Test Report, Build Description Document, User's Manual, Software System User's Manual, Positional

Handbooks, Computer Operation Manual, Program Maintenance Manual, Firmware Support Manual, and Software End-Product Acceptance Plan. Also discussed are software development practices, and methodologies, software design and code inspections, and programming standards. The document and associated Data Item Descriptions appear to have drawn heavily from DOD-STD-2167A and MIL-STD-7935.
INDEX TERMS: System/Software Development, Documentation, Life Cycle Management
OPR: National Security Agency

Table B-1

SOFTWARE POLICIES, STANDARDS, AND GUIDANCE

DOCUMENT ID	Acq	Ada	AIS	C^{3}I	CR	CM	DatMgt	Ed/Trn	Log	Mgmt	Persl	MCCR	Lang	QA	Env	Stds	Dev	Doc	Safe	Sec	T&E	WeaSys
P.L. 81-152	X		X							X												
P.L. 89-306	X		X							X												
P.L. 96-511										X												
P.L. 97-86	X		X																			
P.L. 98-369	X																					
P.L. 98-577	X																					
P.L. 99-591										X												
E.O. 11717			X													X						
NSDD 145																				X		
FIRMR	X				X					X						X						
FAR	X																					
DFAR	X		X		X							X				X				X		
DOD Instruction 3235.1									X					X							X	
DOD Handbook 3235.1-H									X					X							X	
DOD Directive 3405.1		X															X					
DOD Directive 3405.2		X										X	X				X					X
DOD Directive 4100.39			X				X		X													
DOD 4100.39-Manual			X				X		X													
DOD Directive 4155														X								
DOD Directive 4630.5				X												X						
DOD Directive 5000.1	X																X					
DOD Instruction 5000.2	X																					
DOD Directive 5000.3										X		X									X	
DOD 5000.3-Manual-1												X									X	
DOD 5000.3-Manual-3												X									X	
DOD Directive 5000.11							X									X						
DOD Instruction 5000.12							X									X						
DOD 5000.12-Manual							X									X						
DOD Instruction 5000.18							X									X						
DOD Directive 5000.29	X								X	X		X										
DOD Directive 5000.37	X																					
DOD Directive 5000.39	X								X	X												
DOD Directive 5000.43	X																					
DOD Directive 5000.45	X									X												

DOCUMENT ID	Acq	Ada	AIS	C^3I	CR	CM	DatMgt	Ed/Trn	Log	Mgmt	Persl	MCCR	Lang	QA	Env	Stds	Dev	Doc	Safe	Sec	T&E	WeaSys
DOD Directive 5000 49	X																					
DOD Directive 5000 52	X							X														
DOD Directive 5000 53	X							X			X								X			
DOD Instruction 5010.12							X									X						
DOD Directive 5010 19						X																
DOD Directive 5100 30				X																		
DOD Directive C-5200 5				X																X		
DOD Directive S-5200 19				X																X		
DOD Directive 5200 28			X																	X		
DOD 5200 28-Manual			X																	X		
DOD Directive 5215 1				X																X		
DOD Instruction 5215 2																				X		
DOD Instruction 7000 2	X																				X	
DOD Instruction 7000 3	X																					
DOD Directive 7740 1			X				X													X		
DOD Directive 7920.1			X							X												
DOD Instruction 7920 2			X							X											X	
DOD Instruction 7920 4			X							X												
DOD Instruction 7930.1			X					X														
DOD Instruction 7930.2			X		X																	
DOD Instruction 7935.1			X															X				
DOD Directive 7950 1			X		X																	
DOD 7950 1-Manual			X		X																	
MIL-STD-480B						X				X												
MIL-STD-481B						X				X												
MIL-STD-482A						X	X			X												
MIL-STD-483A (USAF)						X										X	X					
MIL-STD-490A	X																					
MIL-STD-499A (USAF)	X									X							X					
MIL-STD-680																X						
MIL-STD-881A										X												X
MIL-STD-882A										X									X			
MIL-STD-1388-1A	X								X													
MIL-STD-1388-2A	X		X						X													
MIL-STD-1397A																X						
MIL-STD-1399C			X		X																	X
DOD-STD-1467 (AR)															X							

DOCUMENT ID	Acq	Ada	AIS	C^{3}I	CR	CM	DatMgt	Ed/Trn	Log	Mgmt	Persl	MCCR	Lang	QA	Env	Stds	Dev	Doc	Safe	Sec	T&E	WeaSys
MIL-STD-1521B (USAF)										X											X	
MIL-STD-1553B					X																	
MIL-STD-1574A (USAF)																			X			X
MIL-STD-1577A								X														
MIL-STD-1589C		X											X									X
MIL-STD-1591				X										X								
MIL-STD-1700	X						X															
MIL-STD-1701					X																X	
MIL-STD-1750A					X																	
MIL-STD-1753													X									
MIL-STD-1777				X																		
MIL-STD-1778				X																		
MIL-STD-1780				X																		
MIL-STD-1781				X																		
MIL-STD-1782				X																		
MIL-STD-1794																	X					X
MIL-STD-1803					X									X			X					
ANSI/MIL-STD-1815A		X	X										X									X
MIL-STD-1838A		X														X						
MIL-STD-1840A							X		X	X						X						
MIL-STD-1862B					X																	
MIL-STD-2001				X														X				
MIL-STD-2076																					X	X
MIL-STD-2084									X													X
MIL-STD-2107														X								
DOD-STD-2167A										X		X					X				X	
DOD-STD-2168														X			X				X	
DOD-STD-5200.28			X																	X		
DOD-STD-7935A			X															X				
MIL-STD-DIF			X													X						
MIL-HDBK-59	X								X	X												X
MIL-HDBK-245B	X																					
MIL-HDBK-286														X			X					
MIL-HDBK-287																	X	X				
MIL-HDBK-334	X													X								
MIL-HDBK-782															X							
MIL-D-28000 - IGES																X						

DOCUMENT ID	Acq	Ada	AIS	C^3I	CR	CM	DatMgt	Ed/Trn	Log	Mgmt	Persl	MCCR	Lang	QA	Env	Stds	Dev	Doc	Safe	Sec	T&E	WeaSys
MIL-M-28001 - SGML																X		X				
MIL-R-28002 Raster																X		X				
MIL-D-28003 - CGM																X		X				
MIL-H-46855B																	X					
MIL-Q-9858A														X								
MIL-S-83490	X																					
AR 25-1			X																			
AR 70-37						X																
AR 70-XX										X		X										
AR-70-1	X				X																	
AR 1000-1	X																					
DARCOM R-70-16					X							X										X
CECOM-R 11-25		X											X				X					X
CECOM-R 11-31					X																	
CECOM-Policy-Compiler Dist.		X						X					X									
CECOM-R (to be assigned)	X											X		X	X							
SECNAVINST 4130.2						X																
SECNAVINST 4200.32																	X					
SECNAVINST 4210.6A	X																					
SECNAVINST 4210.7A	X																					
SECNAVINST 4210.9	X																					
SECNAVINST 4855.1														X								
SECNAVINST 4858.2E														X								
SECNAVINST 5000.1C	X																					
SECNAVINST 5000.2	X																					
SECNAVINST 5000.39A	X								X	X												
SECNAVINST 5200.18							X									X						
SECNAVINST 5200.19							X									X						
SECNAVINST 5200.24							X									X						
SECNAVINST 5200.32										X		X										
SECNAVINST 5200.37	X			X																		
SECNAVINST 5230.3B											X											
SECNAVINST 5230.4			X																			
SECNAVINST 5230.8																X						
SECNAVINST 5230.9A										X												
SECNAVINST 5230.10					X					X												
SECNAVINST 5231.1B	X									X												

DOCUMENT ID	Acq	Ada	AIS	C^3I	CR	CM	DatMgt	Ed/Trn	Log	Mgmt	Persl	MCCR	Lang	QA	Env	Stds	Dev	Doc	Safe	Sec	T&E	WeaSys
SECNAVINST 5231.3A	X									X												
SECNAVINST 5232.1														X								
SECNAVINST 5233.1B																		X				
SECNAVINST 5234.1B													X									
SECNAVINST 5234.2		X											X									
SECNAVINST 5236.1B	X				X																	
SECNAVINST 5236.2A	X				X																	
SECNAVINST 5236.4			X		X																	
SECNAVINST 5237.2					X																	
SECNAVINST 5238.1C					X																	
SECNAVINST 5239.2			X																	X		
SECNAVINST 5420.176					X												X					
SECNAVINST 5420.188B	X									X												
SECNAVINST 5430.98					X					X												
SECNAVINST 5230.11					X					X												
SECNAVINST 10550.4					X																	
OPNAVINST 3960.10C												X									X	
OPNAVINST 4790.2B									X													X
OPNAVINST 5000.42C	X									X							X					
OPNAVINST 5200.28	X									X		X										
OPNAVINST 5239.1A																				X		
NAVMATINST 3960.6B																					X	
NAVMATINST 4130.1A						X																
NAVMATINST 4130.2A						X				X												
NAVMATINST 5200.27A						X			X	X												
NAVSO P-3656	X																					
TADSTAND A												X				X						
TADSTAND B												X				X						
TADSTAND C	X												X									
TADSTAND D					X																	
TADSTAND E												X					X	X			X	
NAVAIRINST 3960.2A										X											X	
NAVAIRINST 4000.14A									X													X
NAVAIRINST 4130.1B						X																
NAVAIRINST 4200.14B	X																					
NAVAIRINST 4275.3E						X																
NAVAIRINST 5100.3B																			X			X

DOCUMENT ID	Acq	Ada	AIS	C^3I	CR	CM	DatMgt	Ed/Trn	Log	Mgmt	Persl	MCCR	Lang	QA	Env	Stds	Dev	Doc	Safe	Sec	T&E	WeaSys
NAVAIRINST 5215.8C						X				X												
NAVAIRINST 5230.5	X									X		X										
NAVAIRINST 5230.6	X					X						X									X	
NAVAIRINST 5230.9									X	X												
NAVAIRINST 5230.11	X									X		X										
NAVAIRINST 5400.14C										X												X
AV-2000A	X																					X
AV-1000A, Sup	X																	X				X
AV-1000B	X																	X				X
SPAWARINST 5200.22	X									X		X										
SPAWARINST 5200.23										X							X					
CNETINST 1500.21								X														
NAVSEAINST 4130.12										X												
NAVSEAINST 4855.25														X								
NAVSEAINST 5450.41B									X													
NAVSEAINST 5450.49									X													
MCO 3093.1C				X												X						
MCO P3900.XX														X			X					
MCBul 3900										X												X
MCO P4105.XX									X													
MCO 4130.2						X																
MCO P4130.8						X																
MCO P5000.10C	X																					
MCO 5000.15	X																					
MCO 5200.23A	X				X					X		X										
MCO 5230.2D																	X					
MCO 5230.15							X															
MCO P5231.1										X												
MCO 5234.4						X																
MCO 5271.1																X						
MCO 5271.2					X					X												
MCO 5271.3					X					X												
MCO P5510.14			X																	X		
MCO 5311.5	X									X												
AFR 57-4									X	X												
AFR 122-9																			X		X	X
AFR 700-4, Vol I	X			X						X												

DOCUMENT ID	Acq	Ada	AIS	C^3I	CR	CM	DatMgt	Ed/Trn	Log	Mgmt	Persl	MCCR	Lang	QA	Env	Stds	Dev	Doc	Safe	Sec	T&E	WeaSys
AFR 700-9, Vol I																X						
AFR 700-26	X									X												
AFR 700-53				X						X												
AFATL Pamp 800-1	X				X					X		X										
AFR 800-2	X									X												
AFOTECP 800 2																					X	
AFR 800-3	X																X					
AFSC/AFLC Pamp 800-5					X					X											X	
ASD Pamp 800-5										X							X					
AFR 800-14	X				X					X							X					
AFSC Pamp. 800-14	X				X					X							X					
AFSC/AFLC Sup 1-AFR 800-14	X									X							X					
AFSC Pamp. 800-43	X									X												
AFSC/AFLC Pamp 800-45					X					X												
AFSC Pamp 800-51										X							X					
SDS Software Policy	X	X				X			X	X			X		X	X	X	X		X	X	
SDIO D 3405	X	X				X			X	X			X		X	X	X	X		X	X	
DCA Memorandum, H102		X											X									
DCA Inst 630-125-1							X									X						
DCA Inst 630-125-2							X									X						
DCA Inst 630-230-9													X			X						
DCA Inst 630-230-17			X															X				
DCA Inst 630-230-19			X																	X		
DCA Inst 630-230-21			X																			
DCA Inst 630-230-23			X						X													
DCA Inst 630-230-28			X							X												
NSAM 81-2	X																X					
NSAM 81-3/MIL-STD-1703 (NS)										X							X	X				

NSAM 81-2

ANNEX C

Current Software Research and Development Efforts

February 9, 1990

CONTENTS

ANNEX C

C. Current Software Research and Development Efforts

Prior to the identification of actions required to improve technology based efforts, one must have an understanding and appreciation for the current DoD software research and development efforts. To date, such collective information has not been readily available, primarily because the DoD does not have a centralized system or office for the strategic planning of software related efforts.

As noted by the Office of Technology Assessment [OTA89], the Deputy Director of Defense Research and Engineering (Research and Advanced Technology) has the responsibility for coordinating the software technology base programs within the Military Departments. However, these programs are coordinated with DARPA's software technology program at one level higher up on the OSD chain. Furthermore, the coordination of the SDI software technology program is accomplished only at the highest level within OSD.

The information provided in this annex represents the current state of software research and development efforts within the DoD, including the Military Departments, Defense Agencies and SDIO. Efforts are listed according to technology area. For each effort cited, the following information is provided: the title of the effort, the objective of the effort, the approach used to accomplish the effort, the payoffs anticipated as a result of the effort, and major fiscal year 1989 accomplishments that resulted from the effort. A financial summary of all of these efforts, on the basis of technology area, is given in Figure C-1.

This annex represents the first consolidated attempt within DoD to identify and categorize all of its unclassified software research and development efforts. Information on any potential classified effort(s) has been excluded from this document in order to allow the widest possible *dissemination to the DoD software community.*

As a result of the comprehensive inputs provided by the participating DoD organizations in the development of the DoD Software Master Plan, the information provided within this annex provides a substantive basis for identifying voids or deficiencies within the overall DoD technology base program. This annex can also be used as the basis for a centralized system for the strategic planning of software research and development efforts. Finally, the information provided in this annex can be used throughout the DoD software community to identify the organizations in which specific efforts are being conducted and from which the resultant technologies can be transitioned into use.

C.1 Software/System Engineering

C.1.0.1 Army: Ballistics Research Laboratory

Title: Tactical Computer Science Technology (7146)

Objectives:

- Investigate, develop, and evaluate innovative technology to help solve critical tactical computer problems.

Approach/Thrusts:

- Develop Tactical Information Distribution Technology with focus on security.
- Develop low echelon and combat vehicle decision aid technology.
- Develop simulations to evaluate Distributed Information and and Decision Aid systems in a total operational context.

Payoff:

- Will minimize low-echelon computer communications.
- Will provide technology for Low-echelon decision-aids.

Major FY89 Accomplishments:

- Firepower Control Simulation for existing and new C^2 systems has been completed and is ready for verification process.

C.1.0.2 Navy: Office of Naval Research

Title: ONR Systems & Software

Objectives:

- Develop formal scientific underpinnings for design and effective utilization of advanced uniprocessor, parallel, and distributed computers

- Develop scientific foundations for the design of correct, efficient, and reliable software

Approach/Thrusts:

- Mathematical foundations of language semantics
- Very high level programming and specification languages
- Experimental software development environments
- Formal system and software verification techniques
- Novel algorithms and techniques for system construction and on-line resource management

Payoff:

- Confidence in software system behavior
- Substantial reduction in testing
- Cost containment through productivity
- Simplification of maintenance and upgrade

Major FY89 Accomplishments:

- Unity parallel program design paradigm (Misra)
- Proof procedure for liveness properties in concurrent computation (Schneider)
- Discovery of link between language and program complexity (Paige)
- Transformational algorithm design and analysis environment (Smith)
- Identification of connections between petri nets and linear logic (Meseguer)
- Graph-theoretic proof procedure for real-time logic (Mok)

C.1.0.3 DARPA

Title: Software Engineering Institute (SEI)

Objective: Advance the state of the software engineering practice to improve the quality of software in mission-critical computer systems. Help DoD and defense industry improve their ability to produce quality software more effectively and predictably by promoting the evolution of software engineering from an ad hoc, labor-intensive activity to a managed, technology-supported discipline.

Approach/Thrusts:

- Software Process.
  - Assist organizations in improving their software process.
  - Develop and transition improved acquisition techniques to DoD.
  - Develop and introduce improved software management methods.
- Software Methods
  - Accelerate the development, introduction, and reduction to practice of methods, tools, and environments that improve software productivity and enhance quality.
- Software Systems
  - Assist the MCCR community in improving the way software is developed for real-time distributed systems.
- Education
  - Increase the number of highly qualified software engineers by rapidly improving software engineering education throughout academia, government, and industry.
- Technology Transition
  - Accelerate the transition and adoption of improved software engineering practice and technology by coordinating and managing transition efforts.
- Ada and STARS Support
  - Remove technical and managerial impediments to the adoption of Ada.
  - Support the DoD software initiative and the STARS Program in technology development and transition efforts.
  - Develop and transition new software engineering approaches and paradigms made possible by Ada language features.

Payoff

- Reduced management and technology risk in acquiring and developing systems that depend on software.
- Enhanced ability of the universities to teach software engineering.
- Enlarged pool of qualified software people available to develop software for the DoD.

Major FY89 Accomplishments

- Expanded the process assessment program to more DoD and defense contractor organizations and expanded assessment of the state of the practice.
- Established an aggressive program to train organizations to conduct self-assessments.
- Applied recent research results on real-time scheduling algorithms to a distributed real-time environment; work demonstrates how to design and implement real-time systems using analytic scheduling algorithms.
- Developed a prototype kernel that supports real-time Ada applications on distributed targets.
- Developed Serpent, a user interface management system that allows integration of new input/output technologies and provides rapid prototyping capabilities.
- Developed a binding between Ada and Structured Query Language (SQL) that facilitates use of Ada with database management systems, thereby enabling use of Ada for C^3I systems.
- Designed and delivered academic courses via video for a professional master of software engineering degree.
- Established a continuing education series, which provides video courses tailored to the advanced level and time constraints of practitioners.

- Established the Computer Emergency Response Team/Coordination Center whose primary purpose is to supplement existing mechanisms for responding to and preventing computer security emergencies. This team/center has been instrumental in thwarting attacks on specific systems and has continued assisting client communities in dealing with security events.

Major Users and Related Activities

- Affiliated with over 40 government organizations and programs including F-16 System Program Office, Maverick Missile Seeker, Granite Sentry, System Program Office for Training Systems, Advanced Field Artillery Tactical Data Systems, Army WWMCCS Information System, Standard Installation Personnel System, Air Defense Initiative, SSN-21 Submarine Integrated Combat System (BSY-2), Ada Joint Program Office, Next Generation Computer Resource, Naval Surface Warfare Center, Naval Air Development Center, Air Force Communications Command, and Jet Propulsion Laboratory. Examples of specific support include:
  - — Granite Sentry - simplified design process and created reusable designs for recurring problems.
  - — Standard Installation Personnel System - developed an interface to permit the use of Ada with database management systems that use SQL.
  - — Navel Surface Weapons Center - provided support in porting Ada programs and supplied benchmarks.
  - — Ada Joint Program Office - provided support for Ada 9x effort.
- 237 industry affiliates including most major defense contractors such as Boeing, Northrop, General Electric, Magnavox, Hughes, Raytheon, TRW, Westinghouse and Rockwell.
- 47 academic affiliates including United States Air Force Academy, The University of North Carolina, Columbia University, University of Maryland, and University of California.
- Related Activities include:
  - — Air Force Systems Command Software Action Team
  - — Joint Integrated Avionic Working Group
  - — Next Generation Computer Resource
  - — Army Software Action Plan
  - — Air Force Communications Command Software Career Management Program

C.1.0.4 DARPA

Title: Software Technology for Adaptable, Reliable Systems (STARS)

Objectives:

- Product-quality Software Engineering Environments (SEE)
- Proof-tested DoD software applications
- Complementary advances in software process, methods, and metrics
- Widely used DoD software component repository

Approach/Thrusts:

- SEE reinforcement of improved process model, emphasizing prototyping, reuse, evolutionary development. Use of model to develop STARS SEEs.
- Definition and use of open-systems SEE interfaces to stimulate tool interoperability and growth of commercial SEE and CASE capabilities.
- Concurrent Service and industry SEE-user involvement to ensure relevance, rapid technology transition.
- Focus on support of Ada and object-oriented development.
- SEE orientation around unified object management capabilities.

Payoff

- Significant improvements in DoD software productivity and quality.
- Reduced time to deply and upgrade DoD software systems.
- Ability to develop more complex software systems with lower risk.

Major FY89 Accomplishments

- Development of initial STARS repository capabilities.
- Software process model improvement
- Identification and analysis of key object-management and Ada open-interface issues.
- Development of Ada-oriented test support capabilities.

Major Users and Related Activities

- Related Activities include: DARPA Arcadia, RADC SLCSE, ASDC DCDS, CECOM DAPSE, Navy ALS-N, NASA SEE

C.1.1 Requirements

C.1.1.1 Army: CECOM, Center for Software Engineering

Title: Requirements Specification Through Prototyping

Objective: Develop software acquisition model will improve the identification and control of requirements and reduce the rework caused by acquisition changes.

Approach/Thrust:

- Develop and document acquisition model.
- Select pilot project and verify model.

Payoff:

- Cost, schedule, and quality improvements by reducing requirements ambiguity and the number of requirement changes.

Major FY89 Accomplishments:

- *An acquisition model for requirement development/control will be defined.*

C.1.1.2 Army: CECOM, Center for Software Engineering

Title: Requirements Engineering Technology

Objective: Investigate feasibility of providing integrated requirements engineering tool set. Also, investigate the application of object oriented technology to requirements engineering and change management.

Approach/Thrust:

- Enumerate requirements for an integration platform
- Shadow effort for a portion of the Limited Operational Capability Europe / Limited Energy Situation Correlation Element, being developed for the Joint Tactical Fusion Program Management Office.

Payoff:

- Improved productivity, software quality, and requirements traceability

Major FY89 Accomplishments:

- New Effort

C.1.1.3 Air Force: Rome Air Development Center

Title: Techniques for System Concept Modeling

Objectives:

- Further refine the idea of conceptual modeling to represent large, complex, real world C^3I systems for the purpose of identifying, analyzing, validating, and documenting key concepts for systems development
- Develop a realistically sized conceptual model, a set of tools and a methodology for using the tools
- Evaluate the effectiveness of these techniques during the pre- requirements phases of the system development life-cycle

Approach/Thrusts: A generic specification of a C^3I system class, such as Air Defense, will be formulated and documented. From this specification, a complex model of air defense will be developed. The model will capture complex behaviors of C^3I systems in terms of objects, relations, attributes, and functions.

The methodology will include guidelines and procedures for developing, refining, and validating sub-models which capture various aspects and behaviors of C^3I systems.

The tools will provide a user interface to a complex knowledge base, which will enable systems analysts to visualize and manipulate the many, complex objects and relationships which constitute C^3I systems.

Payoff:

- Demonstrate that complex problems, such as air defense, can be represented in an automated environment, thereby extending the real world applicability of artificial intelligence techniques in systems development.
- A determination of how well AI techniques scale up in complex domains.
- A prototype tool that will assist the systems analyst in understanding and documenting system behaviors.

Major FY89 Accomplishments:

- FY89 New start

Major Users and Related Activities: The targeted users of this tool are systems analysts involved in C^3I system forecasting and development

C.1.1.4 Air Force: Rome Air Development Center

Title: Conceptual Modeling Via Logic Programming

Objectives:

- Develop a methodology for utilizing a logic programming language to represent a candidate C^3I system and its environment.
- Develop a demonstration prototype.
- Evaluate logic programming's ability to capture C^3I system concepts.

Approach/Thrusts: The methodology will provide guidance in analyzing, developing, and validating conceptual models. Using a selected logic programming language, the methodology will aid in the identification and analysis of critical functions of a target system. Demonstrations and validation of C^3I system concepts will be performed to determine feasibility in terms of technology and performance issues.

Payoff:

- A new approach to system development in the Concept Exploration and Demonstration/Validation phases of the system development life-cycle.
- Demonstrate the application of artificial intelligence techniques to software/systems requirements engineering.

Major FY89 Accomplishments:

- Demonstration prototype tool and documentation delivered.

Major Users and Related Activities: The targeted users of this tool are systems analysts involved in C^3I system development.

C.1.1.5 Air Force: Rome Air Development Center

Title: Requirements Specification Techniques for Heterogeneous Systems

Objectives:

- Investigate and develop a methodology and supporting tool set for specifying, analyzing and validating the requirements and designs of systems whose hardware architectures include both parallel and sequential processing elements.

- Provide a high level graphical user interface for the specification of sequential and parallel processing activities.
- Provide a library of C^3I reusable modules.

Approach/Thrusts: This work will build upon the Very High Level Language tool (Proto) to produce a new tool (Parallel Proto) and corresponding development methodology to address the wide range of issues encountered in the design and development of parallel, distributed real-time C^3I systems. Mechanisms for specifying the scheduling, concurrency, data dependency and synchronization of parallel processes will be designed and implemented. Alternate parallel processing architectures will be simulated allowing performance comparisons to be made.

The output of this task will consist of a requirements specification and validation tool for heterogeneous systems, a methodology for using the tool and a C^3I demonstration exploiting the tool's capabilities.

Payoff: This work will produce a requirements specification and validation tool supporting both sequential and parallel processing elements. The end user is involved early in the requirements validation process and overall software development costs can be reduced.

Major FY89 Accomplishments:

- Contract award was made in September 1989.

Major Users and Related Activities: Parallel Proto is targeted to be used by Air Force organizations who specify and validate the requirements of heterogeneous C^3I systems.

This project is related to the Parallel Evaluation and Experimentation Platform (PEEP) in that future work may involve integration of Parallel Proto into the PEEP.

C.1.1.6 Air Force: Rome Air Development Center

Title: Requirements Engineering Environment Development

Objectives:

- Improve the quality of Air Force Command, Control, Communications and Intelligence (C^3I) system/software requirements by combining and extending three existing requirements analysis tools, the Rapid Prototyping System, the Controlled Requirements Expression Analyst, and the Very High Level Language System Prototyping Tool.
- Provide a common database for data sharing among the three tools and any tools integrated in the future.
- Provide a common user interface to the requirements tools to aid the C^3I requirements analyst.

Approach/Thrusts: This work involves the development of a methodology for using three existing requirements analysis tools. In addition, the methods of each of the tools will be analyzed to identify commonalities, inconsistencies, and enhancements to the tools that may be required. From this, a common database and common user interface will be built for the tools. A key aspect of this work is that the common database and user interface will be sufficiently flexible to allow for the integration of additional tools into the environment.

The output of this task will consist of a requirements engineering environment that will be documented according to DOD-STD-2167A, a methodology for using the environment, and a methodology for integrating new tools into the environment.

Payoff: This work will produce an environment for the specification and validation of the requirements of large and complex C^3I systems in which rapid prototyping plays a significant role. Since the user interface is at a very high level, prototypes of the requirements can be developed rapidly with little or no programming required on the part of the analyst. The end user can be involved very early in the requirements validation process and overall software development costs can be reduced.

Major FY89 Accomplishments:

- Contract was awarded on 27 September 1989.

Major Users and Related Activities: The requirements engineering environment is targeted to be used by Air Force organizations who specify and validate the requirements of C^3I systems. In particular, this project has RADC/OC and NORAD as potential users.

This project is related to the Software Life Cycle Support Environment (SLCSE) in that future work will include the integration of the Requirements Engineering Envvornment Development database into the SLCSE database.

C.1.2 Metrics

C.1.2.1 Army: CECOM

Title: Executive Management of Software (A094 Tactical Software Engineering Technology)

Objective: Develop Army specific software management indicators to provide early, high-level management insight into software development process in order to control the emerging product.

Approach/Thrusts:

- Analyze current experience in use of software management indicators
- Evaluate AMC software management indicators
- Update software management indicators
- Develop guidance for the tailoring and use of software management indicators
- Develop procedures for automated collection and analysis of management indicators

Payoff:

- Validated set of Army management indicators
- Cost-effective application techniques
- Improved management insight into software development

Major FY89 Accomplishments:

- Participated on RADC & SEI Metrics Working Groups
- Initiated evaluation of AMC Software Management Indicators
- Design Metrics report
- Provide consulting service to DoD Organizations

C.1.2.2 Air Force: Rome Air Development Center

Title: SLCSE Project Management System
Objective: To design and implement an automated system to help government project personnel and their contractors plan and track the progress of Mission Critical Computer System (MCCS) software development.
Approach/Thrust: This project will further develop and extend the existing Automated Project Management System specification and implement an advanced prototype capability within the context of the RADC Software Life Cycle Support Environment (SLCSE). This prototype, the SLCSE Project Management System (SPMS), will consist of both MacIntosh-resident and VAX-resident project management tools that are integrated via the SLCSE project database.
Payoff:

- An advancement in the capabilities of software engineering environments.

Major FY89 Accomplishments:

- Draft DOD-STD-2167A documents of the SPMS System Specification (SS) and the SPMS Software Development Plan (SDP) were produced.
- An Ada specification was developed for the Higher-Level Entity-Relationship Interface, which will allow MacIntosh-resident project management tools to up-load and down-load information to and from the SLCSE project database. A DECnet task-to-task communications prototype was also coded in Ada.
- Extensions to the project management subschema of the SLCSE project database were made to support the data items required by Commercial Off-The-Shelf (COTS) project management tools.

Major Users and Related Activities: All users of the SLCSE whose role includes the management of computer software during all phases of the life cycle will benefit from this capability. It will also be integrated with the SLCSE data base and will be able to produce MIL-STD documentation to accompany the entities being managed. These include all types of events from reviews and audits, through baseline capture, and into the production of and support to application software systems being developed using the SLCSE. The effort is related to the planned development of an interface to the SLCSE to accommodate knowledge based software tools and methods.

C.1.2.3 Strategic Defense Initiative

Title: Software Measurement Process
Objective: Develop a software measurement process for the SDS and identify appropriate tools to support implementation.
Approach/Thrusts:

- Define SDS software measurement requirements
- Evaluate current software measurement process
- Evaluate software measurement tools and environments
- Develop an SDS software measurement plan

Major FY89 Accomplishments:

- SDS software characteristics identified and corresponding quality requirements defined
- Thirty six SDS software application domain defined
- Review of existing metrics
- Reviewed extensive list of metric tools and environments
- Determined metric tools applicable to SDS software

C.1.3 Design

C.1.3.1 Air Force: Wright Research and Development Center/Avionics Laboratory

Title: Modular Embedded Computer Software
Objectives:

- Create an integrated design information representation scheme, design methodology, and support tools for distributed, fault-tolerant, multiprocessor avionics software.
- Provide quantitative measures of the impact on system modifications, along with standard consistency checks among components.

Approach/Thrusts:

- Examine current design methods, techniques, and technology.
- Develop a technology to support a system-level design methodology.
- Integrate existing and the newly-developed prototype tools into a system.
- Transition technology to Software Technology Support Center at Hill AFB.

Payoff

- Modified software can quickly be mapped to available computer resources.
- Air Logistics Centers (ALC) will be able to identify impact of changing software on processors, and memory; plus impact on bus utilization.
- Users will be able to determine how software performance affects mission reliability.

Major FY89 Accomplishments

- Prototype tool suite top-level design completed.
- Research phase completed; transition made from research to prototype/exploratory development.

C.1.3.2 DARPA

Title: DARPA Software Design Program

Objectives:

- High performance parallel algorithms suited to emerging parallel and distributed systems for common computational problems.
- Algorithm development and analysis techniques for parallel and distributed systems.
- Language and tool support for the development, analysis, and optimization of parallel software.
- Online management of design and documentation records for large-scale software/hardware systems.
- Use of retained design and documentation records to support rapid adaptability and reuse for systems, interfaces, and components.

Approach/Thrusts:

- Develop solutions to algorithm design problems in areas such as search, discrimination, network reconfiguration, and computational geometry, that have broad Defense-related applications.
- Develop means to retain and apply software design and documentation records that include requirements elements, design documentation, code components, formal annotations and analysis results, test cases and empirical analysis results, metric data, interface definitions, and so on.
- Develop and implement language and tool environments for parallel systems, including analysis, optimization, and refinement tools for scientific applications.
- Enable hybrid parallel language systems that can exploit existing scientific libraries without requiring new code development in the language of the library components (usually Fortran).
- Develop advanced optimization techniques for high performance parallel computers.

Payoff

- Potential order-of-magnitude improvements to systems performance without additional hardware investment as a result of algorithm design improvements.
- Rapidly adaptable software systems.
- Language and associated optimization techniques for parallel scientific and engineering software that can exploit existing software components and libraries.
- Customization of reusable software interfaces and components.

Major FY89 Accomplishments

- Many algorithm improvements, e.g., for high-capacity and fault-tolerant routing in butterfly networks, for improving the efficiency of machine learning, and for solving computational geometry problems applicable to robot movement.
- Very high performance compilation for parallel systems with interprecedural flow analysis and capability for multi-language support.
- Process model for development of large scale Ada trusted systems.
- Prototype system for transformational development and analysis of parallel programs.
- Techniques for inferring type declarations for languages with type systems richer than Ada, including object-oriented systems.
- Initial demonstration of technique for integrating simple reusable components into complex tailored abstract type definitions.
- Preliminary definition of Ada interfaces for MACH.

C.1.4 Development Methodology

C.1.4.1 Army: CECOM

Title: Life Cycle Process (A094 Tactical Software Engineering Technology)

Objective: Define, develop, & document life cycle process models, methods & tools for the creation & evolution of Army software critical systems.

Approach/Thrusts:

- Analyze existing software processes (Government & Industry)
- Develop means for evaluating & selecting software methods & tools
- Develop improved software process models
- Assess software engineering tools
- Investigate software methods, techniques, & tools for improving the software process in all phases of the life cycle
- Explore requirements engineering & rapid prototyping
- Transition and/or export technology improvements & lessons learned

Payoff:

- Evaluation of existing software practices
- Catalog of existing software methods & tools
- Requirements engineering framework, methods, & techniques
- Improved software life cycle models & methods
- Transition into practice via acquisition guidelines, policies, & standards

Major FY89 Accomplishments

- Software Process:

- — Characterized an idealized software development process
- — Defined improved software process models
- — Completed analysis of the requirements definition process
- Software Engineering Disciplines, Methods & Tools:
  - — Updated the software methodology catalog
  - — Developed an approach for evaluating software methods
  - — Developed an approach for assessing software engineering tools
  - — Studied and assessed Cherry's PAMELA2 methods and AdaGraph tool
  - — Improved methodology assessment simulation capability
- Requirements Engineering:
  - — Conducted survey of Requirements Engineering Tools

C.1.4.2 Navy: Naval Ocean Systems Center

Title: Software Technology Project: Processes Task

Objective: Create, define, clarify practical processes relevant to Navy software which can be used to manage and guide the work of software creation and life-cycle support

Approach/Thrusts:

- Advancement of fundamental work (Process; Process Models; Process Representation)
- Forging of agreement and consensus on Primary Life Cycle Model
- Formalization of guidance (standards)

Payoff:

- New Navy life-cycle model applicable to all embedded computer systems

Major FY89 Accomplishments:

- Established initial working version of a process model agreeable to all project participants

Major Users and Related Activities:

- Cooperation with Air Force (RADC) - Technology

C.1.4.3 Air Force: Rome Air Development Center

Title: System Engineering Concept Demonstration

Objective: To demonstrate the concept of an advanced computer-based environment of integrated software tools and methods which supports the Air Force computer-based systems (i.e., software, firmware, and hardware) life cycle.

Approach/Thrust: System requirements/design and software requirements of the system engineering and development environment will be developed and documented. Demonstrations of technologies which are critical to establishing a system engineering and development environment capability will be provided.

Payoff:

- An advance in the system engineering and development state-of-the-art.
- An increase in system development productivity and product quality.
- Specifications which can be used as input to a follow-on advanced development program for a system engineering environment.

Major Users and Related Activities: Capabilities arising from this effort will be used to augment the RADC Software Life Cycle Support Environment (SLCSE) in order to evolve it from a software oriented environment to a system oriented development and support environment.

C.1.4.4 Air Force: Rome Air Development Center

Title: System Engineering Life Cycle Data Model

Objective: To specify a data model of the C^3I system engineering and development life cycle.

Approach/Thrusts: Development of the data model will be driven by pertinent Air Force and DoD system/software development regulations and standards. The data model will be specified using Entity-Relationship technology.

Payoff: A data model which can be incorporated into an advanced system engineering and development environment.

Major FY89 Accomplishments: None - Late FY89 start.

Major Users and Related Activities: The results of this work are applicable to the Software Life Cycle Support Environment (SLCSE) and System Engineering Concept Demonstration efforts.

C.1.5 Software Reuse and NDI Software

C.1.5.1 Army: CECOM

Title: D247 Tactical C^3 Technology Integration

Objective: To conduct development and demonstrations for C^3 System integration efforts and to participate in joint service demos in support of Joint Directors of Laboratories (JDL). Develop and demonstrate common user applications (fiber optic cable system, digital UHF Electronic Counter-Counter Measures technology, multiple channel and cellular technology for Mobile Subscriber Equipment) with emphasis on NDI. Provide users with improved HF/VHF communications. Establish a Technology Assessment Center (TAC) to assess state of the art C^2 technology from a user point of view.

C.1.5.2 Army: CECOM, Center for Software Engineering

Title: Software Reuse (A094 Tactical Software Engineering Technology)

Objective: Identify methods, techniques and tools necessary for the development and use of reusable software and provide guidance on their use in developing Army software critical systems.

Approach/Thrusts:

- Define methodology framework for the development and use of reusable software
- Develop approach for selecting best software reuse approach for Army software critical applications
- Propose changes to existing business practices to facilitate and encourage the development and use of reusable software
- Use proof of concept experiment to test proposed methods, approaches, and concepts

Payoff:

- Reusability guidelines document
- Evaluate technology associated with selected existing library tools
- Develop approach for conducting domain analyses for Army software critical systems
- Develop approach for assessing risk associated with reuse methods used for embedded systems
- Recommend incentives and revised procurement procedures for encouraging and facilitating software reuse
- Provide an approach to interface Ada to SQL

Major FY89 Accomplishments:

- Analyzed relationship between domain analysis and reuse methods
- Conducted in-house experiments using existing CAMP library and tools
- Determined feasibility of application of CAMP to C^3I domain
- Developed concept to conduct domain analysis for Army software critical systems
- Completed Ada/SQL Interface Prototype
- Developed guidelines for evaluation of contractor's Ada Style Guides
- Initiated National Security Industrial Aassociation Software Reuse Study to examine incentives and other non-technical reuse issues

Major Users and Related Activities:

- Direct consulting to Center for Software Engineering who support various systems
- Direct and indirect consulting to Army PMs and PEOs
- Exchange of information with other organizations, e.g. Airmics, SEI, NATO

C.1.5.3 Air Force: Wright Research and Development Center/Avionics Laboratory

Title: 2003 Reusable Ada Avionics Software Packages

Objective: Identify, design, test, implement, and document reusable Ada avionics software life-cycle objects (e.g., algorithms, designs, components, sub-systems,and associated tests and documentation) for use with real-time embedded computer software. Define the domain, develop a library structure, and develop, build, and demonstrate reusable Ada software parts for avionics applications.

Approach/Thrusts:

- Reusable real-time avionics system
- Algorithms, designs, sub-systems, test cases
- Multi-faceted component classification schema
- Configuration management system
- Expert-based Library manager

Payoff:

- Accelerated design & development time
- Cost reductions
- Enhances reliability

C.1.5.4 Air Force: Rome Air Development Center

Title: Reusable C^3I Specifications

Objective: To develop a methodology and prototype support tools for reusing requirement and design specification components during the requirements analysis and preliminary design phases of the C^3I system life cycle.

Approach/Thrusts: A methodology for the reuse of specification components based on customizing reusable modules, interactive development and interpretive execution will be defined. A library for storing the reusable components will be designed and implemented using object oriented data base techniques. Strategies for classifying the components, searching for them and integrating them into existing specifications will be defined. Tools to support these reusability activities will be designed and developed with uniform user access interfaces. The library mechanisms and support tools will be used to construct a set of reusable components for a selected Air Force C^3I problem.

Payoff: This effort will produce a set of reusable specifications for an important class of C^3I problem along with the techniques and tools to maintain these specifications. These specifications will then serve as a model for future efforts to create reusable components and can themselves be reused in future projects.

Major FY89 Accomplishments: During FY89 a demonstration of the support tools environment for specifying concurrent processes, necessary for the specification of C^3I systems, was successfully conducted. Also, the C^3I problem of multiple target tracking within an air defense system was selected as the subject for the library of reusable specification components. Identification and development of reusable library components was begun.

Major Users and Related Activities: This effort will provide a requirements validation capability based on specification and prototyping. It will be integrated with a requirements analysis capability and other prototyping tools to form a requirements engineering workstation. This workstation will be coupled into the Software Life Cycle Support Environment (SLCSE) to enhance its requirements engineering capabilities. Several AFLC Air Logistics Centers are users of the SLCSE. The RADC/OC User Friendly Radar Simulation System program and the NORAD Granite Sentry program office are potential users of the requirements engineering workstation.

C.1.6 Quality Methods

C.1.6.1 Air Force: Rome Air Development Center

Title: Quality Evaluation System (QUES)

Objectives:

- Develop new software quality metrics to augment the existing software measurement and assessment framework.
- Provide automatic data collection and analysis to aid requirements engineers, designers, programmers, test personnel, test managers, and acquisition managers in assessing software quality to determine compliance with required/desired software quality goals.
- Provide graphical and textual reports for acquisition personnel for them to easily assess the quality of the products that are being developed in accordance with DoD-STD-2167A.

Approach/Thrusts: It is generally agreed that the results of analyzing quality solely at the end of the life cycle are unsatisfactory. Systems thought to be correct, reliable, maintainable, expandable, and purchased as such by the Air Force, may suddenly produce incorrect results, no results, or may be very difficult to modify. Because software is a major element in most military systems, substantial resources are currently expended by the Air Force to analyze and improve software quality after the system is delivered. This effort will:

- Analyze the RADC Software Quality Framework to determine additional data which would be important to requirement engineers, designers, programmers, test personnel, test managers, and acquisition managers.
- Develop QUES using an incremental build and object-oriented approach.
- Integrate QUES with the RADC's Software Life Cycle Support Environment (SLCSE) for support of all life cycle phases in accordance with DoD-STD-2167A.
- Interface with the specification tool called Assistant for Specifying the Quality of Software (ASQS) for a complete specification, evaluation, and assessment process.

Payoff:

- A highly productive tool to aid in the development of quality software for future systems.
- A tool which supports the entire software development life cycle allowing for traceability of software quality requirements.
- Support for Ada and Fortran system and software development.

Major FY89 Accomplishments:

- Completed Build 1 of 3 builds. Build 1 contains the basic windowing, database and user interface capabilities.
- Software Requirements Specification delivered.

Major Users and Related Activities: This project is targeted for System Program Offices to aid them in determining the quality of a software system and its responsiveness to mission/user requirements.

C.1.6.2 Air Force: Rome Air Development Center

Title: Software Quality Automated Method Validation

Objective: The objective of this effort is the validation of the RADC software quality measurement methodology and tools that support the methodology. The goal of the methodology is to enable an acquisition manager to acquire a software product(s) which satisfy user quality needs (i.e., reliable, supportable). There are three major parts to the process: 1) specifying software quality requirements, 2) evaluating achieved software quality levels, and 3) assessing compliance to the required quality levels.

Approach/Thrusts: The Assistant for Specifying the Quality of Software (ASQS) is an expert system which aids an acquisition manager in specifying software quality goals. ASQS automates the specification and assessment parts of the methodology. ASQS will provide the "why" and "how" the factors are chosen based on the acquisition managers inputs. ASQS removes the factors, criteria and metrics that are not applicable to a particular project. The QUality Evaluation System (QUES) is a tool which evaluates the quality of the products at each phase of the life cycle in accordance with DoD-STD-2167A. QUES automates the evaluation part of the methodology. QUES gives insights into how the products are meeting quality requirements and indicates what factors and where the deficiencies lie in the products. QUES takes specification of quality from ASQS, does the evaluation based on the inputs from ASQS and returns the evaluation results to ASQS for assessment of compliance to the specified quality requirements. The tools and framework, on paper and in the laboratory, have proved promising but need to be applied to a project(s) that are either on going or just starting.

The scenario of using ASQS and QUES is to provide continuous feedback on the overall assessment of the product(s) being developed and to insure that quality is designed into the software rather than merely tested for. This will provide a quality product and a cost savings not only in development but in the operation and support phases. The framework and tools will be applied to a RADC program(s) that is on going and/or just starting. The tools will be applied to the project(s) to specify goals, evaluate the products and assess the results versus the success of the project(s). In applying the tools and framework, any inconsistencies, ambiguities or omissions will be collected and used to improve the knowledge base of ASQS, improve utility and robustness of ASQS and QUES and help in establishing the applicability of the RADC Software Quality Framework and supporting tools. The ASQS will be used to specify, up front, the quality goal requirements and assess the evaluation of the products. The QUES will be applied to all the Full Scale Development phases collecting data on the products and keeping track of problems with the usage of the tool and framework. From the application of the tools and framework to the project(s), we should be able to gain insights into the applicability of the methodology, tools, and framework.

Payoff:

- Validation of the software quality framework and supporting tools.
- Insights into the applicability of such a technology on real world systems to insure they reliable and supportable.

Major FY89 Accomplishments: FY90 New Start

Major Users and Related Activities: This is an application of a technology that has matured to a point that is ready for application to a system for evaluation and validation of this technology.

DoD will the major user and will via of the technology, produce systems that are not only cost effective and on time but do what they were built to do.

C.1.6.3 Air Force: Rome Air Development Center

Title: Software Quality Laboratory

Objective: The objective of this effort is to assist industry defense contractors in the application of the RADC Software Quality Framework (as defined in RADC TR-85-37, 3 Vols) and supporting tools (Assistant for Specifying Quality Software (ASQS) and QUality Evaluation System (QUES)) to actual software development projects in order to evaluate the Framework's theoretical foundation and its impact on development cost and product quality.

Approach/Thrusts: The approach is to fund a general support contractor to assist industry defense contractors in the specification and evaluation of software quality factors on actual programs, to assist them in the application and evaluation of software quality technology, and to provide a means for technology transition of software quality tools and methods.

Payoff: The payoff expected from this effort is an improvement in defense system software quality with corresponding reductions in software development and support costs.

Major FY89 Accomplishments: FY90 New Start

Major Users and Related Activities: The major users of the results of this project are U. S. defense contractors and defense system program offices.

This effort is related to the Software Quality Automated Method Validation project.

C.1.6.4 Air Force: Rome Air Development Center

Title: Software Quality Process Improvement

Objective: The objective of this effort is to define a total quality control methodology and implementation approach for software acquisition, development and support and fold the RADC Software Quality Framework and AFSC Management and Quality Indicators into the methodology. The RADC Framework is an product-specific approach to building quality into software. The total quality control approach relies on the elements of the process to achieve quality. Therefore, the objective of this effort is to develop a combined product and process approach to software development, acquisition and support.

Approach/Thrusts: The first task of this effort is to review the RADC Software Quality Framework, AFSC Management and Quality Indicators. The second task is to review and summarize aspects of total quality control that are candidates for application to the acquisition, development and support of AF software systems. The third task is to merge the first two tasks into a methodology for the total quality control of software. The approach to developing the methodology considers the following strategies:

- Total Quality System: This review will look at a systems approach to quality, establishing a quality system and quality cost.
- Management Strategies for Quality: Management strategies for quality are concerned with organizing quality and ways of achieving a total commitment to quality.
- Engineering Technology for Quality: This review will focus on quality engineering technology, process-control engineering technology and quality engineering equipment engineering technology.
- Statistical Technology for Quality: The statistical technology commonly used in TQC include the frequency distribution, control charts, sampling tables, special methods and reliability modeling.
- Applying Total Qualitv Control: This task will develop an approach for applying the methodology to the development, acquisition and support of select AF systems.

Payoff: The payoff expected from this effort is improved methods for the acquisition, development and support of DoD software systems. These improvements should lead to the eventual reduction in software development and maintenance costs with improvement in its quality.

Major Users and Related Activities:

- The major users of the methodology will be industry defense contractors and government system program offices.
- A related activity is the DoD Total Quality Management initiative.

C.1.6.5 Air Force: Rome Air Development Center

Title: Language Sensitive Quality Editor (LSQE)

Objective: Develop a language sensitive quality editor (LSQE) software tool capable of assessing, in near real time, the realization of established software quality goals.

Approach/Thrusts: This effort, based on the concept of a language sensitive editor, will develop a knowledge-based approach for storing the editor's metric rules and guidelines, for explaining quality deficiencies. and providing advice on how to improve the quality of the product.

Using the editor, quality analysis will be available in a continuous form of interactive feedback during development. The editor will be integrated into the RADC Software Life Cycle Support Environment (SLCSE). The SLCSE is a computer-based environment which supports the development and post deployment support of mission critical computer systems (MCCS) software in accordance with DOD-STD-2167A.

The Phase II product will be a production quality Ada language sensitive editor that supports both Ada design and coding.

Payoff: The greatest payoff is that quality deficiencies detected early are much easier to correct than those detected after the software is complete. Significant benefit will be realized by this tool's influencing design decisions as they are made - thus avoiding the high cost of making changes later.

Major FY89 Accomplishments: Phase I was successfully completed during FY89. Phase I established the feasibility of this project. The Phase II (actual development) effort commenced 28 Sep 89.

Major Users and Related Activities: The LSQE will have potential for immediate application on software development efforts in both commercial and Government domains (Air Force, Army, Navy, SDIO, NASA).

At the completion of the Phase II SBIR the LSQE will be initially marketed to Ada software developers; however, the underlying technology is language independent. The LSQE's technology could be applied to other languages used in the software life cycle with similar benefits.

A Phase III follow-on is anticipated for FY92 and beyond pending successful completion of Phase II. Phase III will pursue commercial development.

C.1.6.6 Air Force: Rome Air Development Center

Title: System Quality Attributes

Objective: Investigate the relationships between system quality attributes and software quality attributes and develop estimates of degrees of dependence between them for each of the five (5) mission areas (i.e. Armament, Avionics, C^3, Missile/Space, and Mission/Force Management). The five mission areas are consistent with those of the Software Test Handbook developed by General Research Corp. (RADC-TR-84-53, Vol II of two, dated March 1984).

The proposed effort will investigate for example: a system quality attribute, such as availability, which could translate into needs for software quality attributes, such as reliability and maintainability. The software reliability of the Armament area may be required to a lesser, equal, or greater degree than that of the Missile/Space area for the same level of required system availability.

The information developed will be used to upgrade the mission area generic functional decomposition reference models, knowledge base, and rule sets found in the ASQS and the RADC Software Quality Framework.

Approach/Thrusts: This effort will be accomplished under the Expert Science & Engineering Program by Rochester Institute of Technology.

The generic architectures and analyses for the five (5) mission areas being developed by Advanced Technology Inc. for ASQS will be examined. Upon completion of this first task, system quality attributes for each mission area will be established. Criteria will be developed for relating system quality attributes to software quality attributes and also for estimating relative dependencies between them. A dependency matrix for each mission area will then be developed for subsequent translation and implementation into ASQS as refinements to rules. Also upon completion of Task 1, the ASQS information sets for each software quality factor developed by DRC ?? will be updated for subsequent implementation into ASQS.

The last task will be to investigate the impact the previous tasks have on the framework and the weighting and scoring schemes used by ASQS to establish software quality factor requirements.

Payoff:

- Establishment of relationship between system quality attributes and software quality attributes.
- Provide ideas and indicate gaps where in the software quality methodology needs improvement.

Major FY89 Accomplishments:

- FY90 New Start.

Major Users and Related Activities: Results from this work will help in enhancing the software quality methodology and framework.

C.1.6.7 Air Force: Rome Air Development Center

Title: Software Quality Specification Automated Assistant Enhancements

Objective: To develop an automated knowledge base system, called the Assistant for Specifying the Quality of Software (ASQS), to assist acquisition managers specify software quality requirements

Approach/Thrusts: A knowledge base system will be used to translate answers by an acquisition manager to questions about his project provided by ASQS during a consultation session into a tailored software quality specification. It will be designed so that the specification can evolve from a preliminary to a final specification as more information surfaces.

It will be designed for the five (5) mission areas: Armament, Avionics, C^3, Missile/Space, and Mission/Force Management. Emphasis will be placed on rule development for C^3 (Intelligence) and Missile/Space (Satellite). whose generic case models are being developed under another PE 63728F contract, "Specification Assistant Mission Area Generation".

Documentation will be in accordance with DOD-STD-2167A.

Payoff:

- Specification of software quality requirements by acquisition managers by translation from terminology related to project characteristics and requirements.
- Consistent realistic consideration of software quality requirements starting at concept exploration.
- Cost saving realized by automating a complex labor intensive software quality specification process.

Major FY89 Accomplishments:

- Feasibility model has been demonstrated on a Xerox workstation using the EMYCIN knowledge base.

Major Users and Related Activities: The ASQS will be transitioned to the Software Quality Lab proposed for start in FY90 under this program element. A demonstration will be conducted on a Satellite or Intelligence program in conjunction with the QUality Evaluation System (QUES), also being developed under this program element, to prove feasibility of the total semi-automated specification and evaluation method. Successful demonstration will lead to expanded ASQS rules development for the remaining mission areas.

C.1.6.8 Air Force: Rome Air Development Center

Title: Specification Assistant Mission Area Generation

Objective: To develop generic functional decomposition models for the five mission areas (i.e. Armament, Avionics, C^3, Missile/Space, and Mission/Force Management) to assist software acquisition managers develop a software quality specification for their individual projects. Emphasis will be placed on development of the C^3 (Intelligence) and Missile/Space (Satellite) areas.

Approach/Thrusts: The five mission areas will be decomposed into functions. Rules will be developed for acquisition concerns in the C^3 (Intelligence) and Missile/Space (Satellite) areas as they relate to the 13 software quality factors of the RADC Software Quality Framework. The method to be used for developing the rules will be documented so it can be used in the future for rule development in the remaining areas.

Payoff:

- Specification of software quality requirements by acquisition managers by translation from terminology related to project characteristics and requirements.
- Consistent realistic consideration of software quality requirements starting at concept exploration.
- Cost saving realized by automating a complex labor intensive software quality specification process.

Major FY89 Accomplishments: The final technical report is being prepared by the contractor for review by RADC.

Major Users and Related Activities: The results of this work will be integrated into the Assistant for Specifying the Quality of Software (ASQS), a knowledge base tool also being developed under this program element, by the ASQS contractor. A demonstration will then be conducted on a Missile/Space (Satellite) and/or C^3 (Intelligence) program in conjunction with the QUality Evaluation System (QUES) ,also being developed under this program element, to prove feasibility of the total specification and evaluation method which was previously developed and developed as a manual process under this program element.

C.1.6.9 Air Force: Rome Air Development Center

Title: Software Quality Methodology Integration

Objectives:

- To perform an analysis and investigation which integrates the Assistant for Specifying the Quality of Software (ASQS) and the QUality Evaluation System (QUES) into the software quality and evaluation methodology.
- Investigate the issues related to modifying the RADC software quality framework to support Object Oriented Design (OOD).

Approach/Thrusts: The manual methodology approach found in the Specification of Software Quality Attributes guidebooks will be reviewed. Evaluations of ASQS and QUES capabilities will be performed as they relate to a unified integrated methodology. The potential for enhancing the quality framework by examining the characteristics of OOD will be performed and recommendations will be made for modifying framework criteria and metrics, as required.

Payoff:

- A semiautomated method for specifying and evaluating software quality on major software developments.
- Expanded applicability of the RADC Software Quality Framework to include OOD developments.

Major FY89 Accomplishments: The contractor has reviewed the Specification of Software Quality Guidebooks, which were developed previously under this program element and has begun review of preliminary documentation prepared for ASQS and QUES. Review of the existing RADC Software Quality Framework also is completed.

Major Users and Related Activities: The results of this work will become the foundation for training to be accomplished under the Software Quality Laboratory program scheduled for start in FY90 under PE 63728F.

C.1.6.10 Air Force: Rome Air Development Center

Title: System for Software Engineering Capability Enhancement

Objective: The objective of this effort is to demonstrate that a system can be developed which will take as input the results of an SEI capabilities assessment and produce specific detailed recommendations which will, if followed, improve the level of software development capability for a development organization.

Approach: The approach is comprised of three tasks:

1) Preparation of a preliminary design for the automation of the scoring procedure of the SEI assessment methodology

2) Preparation of a preliminary database design for the information related to the specific areas of inquiry contained in the SEI's assessment document. This will contain information on software tools, managerial and organizational structure, as well as training and technology management.

3) Condense, categorize and organize the data gathered in task two into a database design which, when combined with the scoring procedures of task one will yield a detailed set of recommendation in the above areas based on the set of specific responses to the assessment methodology.

Payoff: The payoff is to design a tool which, when built, will provide significant assistance to government and industry alike in determining shortfalls in software engineering capabilities which would be applied to defense system software acquisitions. This tool would also help both government and industry by indicating how to remedy those shortfalls with the most cost effective and productive technologies.

C.1.7 Languages

C.1.7.1 Air Force: Wright Research and Development Center/Avionics Laboratory

Title: 2003 Common Ada Run-Time System

Objective: Design, build, and demonstrate a common Ada run-time system for embedded real-time applications.

Approach/Thrusts:

- Baseline the requirements.
- Design the system to provide common interfaces, features, services, and options to avionics software developers and compiler designers.
- Implement a common run-time system for two compilers targeted to different processors.

Payoffs:

- Portability of Ada run-time systems among different target computers with predictable performance.
- Reduced time in developing real-time software.
- More reliable real-time avionics software for heterogeneous distributed-processor architectures.
- Predictable run-time performance from complex architectures using disparate processors.

Major FY89 Accomplishments:

- New start in FY89.

C.1.7.2 Ada Joint Program Office

Title: Ada 9X Project

Objective: To revise ANSI/MIL-STD-1815A to reflect current essential requirements with minimum negative impact and maximum positive impact to the Ada community.

Approach/Thrusts:

- Revise ANSI/MIL-STD-1815A
- Obtain adoption by: DoD, ANSI, ISO and NIST
- Update Ada compiler validation capability
- Update Ada compiler evaluation capability
- Recommend transition policy/procedures
- Develop education/training program
- Develop language long-term maintenance plan

Payoff: The result of these related efforts will yield a common computer programming language for DoD which will be even more versatile than the standard currently in place and will meet the processing and application performance requirements of tomorrow's highly sophisticated weapons systems.

Major FY89 Accomplishments:

- Collection of revision requests from worldwide Ada community
- Establishment of reviewer organizations and staffing
- Preparation of contract request for proposal
- Publication and dissemination of public reports on progress and planned activity

C.1.7.3 Ada Joint Program Office

Title: Ada Technology Insertion Program (ATIP)

Objectives:

- To encourage the use of Ada in the modification and development of weapons systems. To establish a track record of projects that demonstrate overcoming technical barriers in specific Ada implementations.
- To coordinate Ada education and training within the DoD. To ensure that DoD Ada education and training are of the highest quality taught from a software engineering perspective. To promote Ada in university and academic curricula.

Approach/Thrusts:

- Provides one year of cost shared funding for specific projects that can demonstrate results in overcoming technical barriers to the use of Ada in that time frame, with the intent to disseminate to the Ada community the lessons learned and provide resulting Ada code for reuse. Continue to develop criteria that will better enable technical evaluation of future projects.
- Organizes an annual symposium on effective methods of teaching software engineering and Ada. Conducts joint service advance training sessions in Ada for DoD personnel. Coordinates service efforts to start or increase Ada classes in their curricula. Visits colleges and universities to introduce Ada to curriculum developers.

Payoff:

- The technology insertion projects funded in FY89 included aircraft, submarines, communications, automatic test equipment, command/control, jammers, radars, tanks, ships, simulators, torpedos, missile artillery, air-to-surface munitions, and landmines. The ATIP funds were aimed at research in real-time systems, machine independence, 32-bit avionics computer transition, support environments, reuse, and large systems.
- The DoD education community is adopting Ada for its curricula. For example, the Naval Academy now uses Ada as its base language for midshipmen. Results from a survey of past Ada Software Engineering, Education, and Training Symposium attendees will be published prior to the end of the fiscal year.

Major FY89 Accomplishments:

- Fifteen technology insertion projects were funded in FY89.

- The 1989 Ada Software Engineering, Education, and Training Symposium was attended by 200 DoD, academic and industrial trainers and educators; a figure that is up 30% from last year.

C.1.7.4 Ada Joint Program Office

Title: Ada Validation and Control

Objective: To provide a means by which compiler purchasers may be assured that Ada compilers conform to the standard Ada in all respects, and introduce neither super-sets nor sub-sets of the standard. This is of particular importance to the DoD acquisition process.

Approach/Thrusts: To validate Ada compilers through the use of a suite of tests developed to exercise the compilers to determine conformance with the standard. The validation is performed by Ada Validation Facilities located at Wright-Patterson AFB, Ohio; National Institute of Standards and Technology, Gaithersburg, Maryland; and three foreign facilities in England, France and West Germany. The validation capability is maintained by the Standard Languages and Environments Division at Wright-Patterson AFB, Ohio. The Institute for Defense Analyses provides technical guidance as the Ada Validation Organization.

Payoff: Both the DoD acquisitions organizations and commercial software houses have accepted the AJPO validation seal as a primary criteria for purchasing an Ada compiler. No super-sets or sub-sets of Ada have been entertained by compiler vendors. The number of vendors participating in the validation process has risen from 10 in January 1986 to 52 in June 1989. The number of validated compilers has risen from 3 in December 1983 to 194 base compilers in June 1989.

Major FY89 Accomplishments:

- Released Version 1.10 of the Ada Compiler Validation Capability
- Pre-released Version 1.11

C.1.7.5 Ada Joint Program Office

Title: SQL Ada Module Extensions (SAME)

Objective: To provide a standard interface between Ada application programs and commercial off-the-shelf (COTS) Structured Query Language (SQL) database management systems.

Approach/Thrusts:

- Built on the ANSI Standard Module Language
- Based on the principles of abstraction and encapsulation
- Separates Ada application code from SQL database code
- Enhances visibility and control of the application database interface

Payoff: Provides mechanism for database applications written in Ada to utilize SQL data base management systems, thereby eliminating a major barrier to the use of Ada in such systems.

Major FY89 Accomplishments:

- Demonstration of SAME implementation on four architectures
- Initial public report issued on efforts
- Development of a prototype SAME Processor for automated support
- Changes recommended to ANSI on the embedded SQL for Ada were accepted

C.1.8 Development & Support Environments

C.1.8.1 Army: AIRMICS

Title: DY10-02/Software Engineering

Objective: To develop conceptual and prototype components of a software engineering environment aimed at reducing software life cycle unit cost, increasing the productivity of software support, and increasing the software quality of components, systems, and products delivered.

Approach/Thrusts:

- Address software development management
- Software reuse and metrics
- Software maintenance
- Prototype Ada Programming Support Environment for Information Systems Engineering Command

Payoff:

- Reduce life cycle costs
- Productivity improvements
- Quality improvements
- Production of systems which are usable, reliable, and maintainable
- Improve software development management (tools, techniques)

Major FY89 Accomplishments:

- Management of Software Development
  - Obtained sponsorship
  - Obtained external funds for qualitative assessment
  - Completed assessment orientation training
  - Software Engineering Research Center participation
- Software Reuse and Metrics
  - Hold workshop in Atlanta
  - Perspective on software reuse (report)

  - — Prototype software delivered for test and experimentation
- Software Maintenance
  - — Software tools for software maintenance (report)
- APSE Prototype
  - — Participate in Ada 9X
  - — Literature search
  - — Secure hardware for Proof-of-Principle Demonstration
  - — Initiate Tools vs. Methods Study

C.1.8.2 Army: Strategic Defense Command

Title: Advanced System Engineering Environment Development
Objective: To develop a computer assisted systems engineering environment which will provide the tools and environment necessary to produce the massive quantities of SDS software.
Approach/Thrusts: Provide an early System Engineering Environment (SEE), establish trusted software standards and methodology, and integrated case tools that allow for fast prototyping and software development.
Payoff:A software development environment that will allow for the producibility of tested, trusted, and affordable software for the SDS.
Major FY89 Accomplishments:
Major Users and Related Activities:

C.1.8.3 Navy: NAVAIR

Title: NAVAIR Software Engineering Environment
Objectives:

- Reduce cost of software maintenance
- Increase quality and productivity
- Protect Government's interest
- Ensure competition
- Basis for common metrics

Approach/Thrusts:

- 3 phases
  - — Common Tool Set
  - — Integrate tools
  - — Full spectrum integrated SEE
- Utilize Best-of-Industry (VMS/UNIX/MS-DOS)
- Embrace evolving industry software standards (*POSIX*)
- Transition to support new DoD standards

Payoff:

- Avoid uncontrolled proliferation of tools
- Reduce cost of acquiring software tools
- Able to competitively procure "Best of Industry"
- Rapid introduction of a common tool set
- Promote tools to enable competitive procurement of weapon system software updates and PDSS support

Major FY89 Accomplishments:

- Defined a common tool set to cover major life cycle needs
- In process of competitively procuring tool set

Major Users and Related Activities:

- NADC: Naval Air Development Center
- NWC: Naval Weapons Center
- NADEP: Naval Avionics Depot, North Island
- NSTC: Naval Systems Training Center
- NOSC: Naval Ocean Systems Center
- NAC: Naval Avionics Center
- NATC: Naval Avionics Test Center
- PMTC: Pacific Missile Test Center

C.1.8.4 Navy: Naval Ocean Systems Center

Title: Software Technology Project: Methods and Tools Task
Objective: Find, link, and apply disciplined approaches (methods) combined with automation (tools) to support and increase the effective application of processes needed to create and maintain quality Navy software.
Approach/Thrusts:

- Support for evolving Primary Life Cycle Model (Requirements, Design, Documentation)
  - — Technology acquisition
  - — Technology integration using an existing "tool integration framework"
- Technology for evaluation of software (Performance Assessment, Testing, Verification)

Payoff:

- Methods and tools that support life-cycle model

Major FY89 Accomplishments:

- Completed specifications and technology assessment for Requirements Toolset
- Completed POD development

Major Users and Related Activities:

- Cooperation with Air Force (RADC) - Technology

C.1.8.5 Navy: Naval Ocean Systems Center

Title: Software Technology Project: Applications Task

Objective: Demonstrate that "generic technology" can be synthesized to meet needs of specific Navy applications and transmitted to operational use.

Approach/Thrusts:

- Technology assembly/integration experience
- Project specific transition experience
- Showcase demonstration
- Transition to policy and standards organizations

Payoff:

- Significant application and demonstration of technologies
- Recommended upgrade to DoD 2167

C.1.8.6 Air Force: Wright Research and Development Center/Avionics Laboratory

Title: Automatic Programming Technologies for Avionics Software

Objective: To develop an Automatic Programming system geared to provide high-level automated software specification/design capabilities with respect to the generation of real-time, clock-driven, avionics applications such as distributed processing, tasking, cyclic executive, etc.

Approach/Thrusts:

- Automated programming environment for developing real-time Ada avionics software
- High-level graphical specification and design capability
- Avionics system description language
- Automated design synthesis capabilities

Payoff:

- Substantially improved software development efficiency
- Enhanced software quality
- Graphical (iconographic) design capability
- Ease-of-use-requiring minimal training

C.1.8.7 Air Force: Wright Research and Development Center/Avionics Laboratory

Title: Interactive Ada Workstation

Objective: Improve Ada programming productivity through the use of interactive software technology, and demonstrate this capability through a series of prototypes.

Approach/Thrust:

- Formalization of Buhr diagrams, state machine diagrams, decision tables, and truth tables to enable Ada to be generated from the diagrams.
- Demonstration of feasibility of instantaneous syntax checking of the complete Ada language indepen ـnt of user edits.
- Demonstration of feasibility of incremental semantic (compile as you type) checking with virtual real time feedback to the user for a major subset of Ada.
- Demonstration of an abstract language model for supporting incremental semantic analysis covering the entire Ada language.

Payoffs:

- Greater productivity of Ada programmers.
- More reliable Ada code.
- Less expensive software production.
- More maintainable software designs.
- Reusable Ada code creations.
- Improved ease of programming in Ada.

Major FY89 Accomplishments:

- Technology has been developed, proven, and transitioned to industry for application to DoD programs.
- First tools using this technology are available as commercially-supported products.
- These tools (under the commercial TEAMWORK/ADA trade name) have been specified for use in several major DoD programs including the Space Station Environment (Lockheed), the Joint Interactive Avionics Working Group (for Advanced Tactical Fighter, Army Light Helicopter Experimental (LHX), and ATA avionics), and the STARS program (Boeing).

C.1.8.8 Air Force: Wright Research and Development Center/Avionics Laboratory

Title: Ada Compiler Evaluation Capability (ACEC)

Objectives:

- Compare the performance of several Ada compiler systems.
- Isolate strong and weak points of a specific system.
- Determine significant changes between releases of a specific compiler.

- Predict the performance of differing Ada Design approaches.

Approach/Thrusts:

- Develop a test suite, support tools, and documentation to allow Ada compilation system performance comparisons.
- Assess the usability of Ada compilation systems with respect to the program library system, the symbolic debugger, and the diagnostic messages.
- Measure the following test attributes: compile time efficiency, execution time efficiency, and code expansion.

Payoffs

- Ability of DoD to scientificallyl select the best Ada compiler for its programs.
- Promotion of higher quality compilers for Ada real time applications.

Major FY89 Accomplishments:

- Version 1 has been released to the RADC Data and Analysis Center for Software. It is available to all of DoD and its contractors.
- More than 13 compiler vendors use the ACEC to evaluate their products.
- Formal Qualification tests for version 2.0 of the ACEC were completed in Dec 89.

C.1.8.9 Air Force: Rome Air Development Center

Title: Integration of Knowledge-Based and Conventional Tools

Objectives:

- Investigate and determine the potential application of knowledge-based technology and its benefits to the processes and methods supported by conventional software life cycle development tools.
- Specify potential approaches to augmenting and enhancing conventional tools with an effective knowledge-base and associated knowledge-based software.

Approach/Thrusts: This effort will investigate knowledge-based technology and its potential application to the RADC Software life Cycle Support Environment (SLCSE) and its associated software development tools. The SLCSE is a computer-based environment which supports the development and post deployment support of mission critical computer systems (MCCS) software in accordance with DOD-STD-2167A.

Multiple approaches to the knowledge-based enhancement and augmentation of conventional software tools will be defined and two prototype software tools applying knowledge-based technology will be developed.

The products of this effort are a series of technical reports including a 5 Year Plan for integration of knowledge-based technology into the SLCSE toolset.

Payoff: There are a number of potential benefits to be gained by the insertion of knowledge-based technology into existing conventional tools. Potential benefits include:

- improved productivity, enhanced reliability and maintainability
- reduction of decision making and other tool interaction by anticipating tasks to be performed
- reduction in personnel and level of expertise required to use the tools

Major FY89 Accomplishments:

- An Interim Technical Report for Task II (Analysis of the RADC SLCSE) was completed.
- The software demonstration prototypes were selected and initiated. Two prototypes are being developed: 1) an intelligent change management system, 2) and intelligent database schema editor.

Major Users and Related Activities: A follow-on to this effort will design and implement a subset of the tools identified in this effort for knowledge-based technology insertion.

C.1.8.10 Air Force: Rome Air Development Center

Title: Software Life Cycle Support Environment (SLCSE)

Objective: To develop a computer-based environment of software engineering tools and methods capable of effectively supporting the development of Air Force mission critical computer system (MCCS) softv are.

Approach/Thrust:

- Initial requirements and high level design were based on the results of Contract F20602-84-C-0120 entitled "C^3I Support Environment Def nition".
- Development of eight (8) incremental builds, constituting three (3) formal version deliveries.
- A majority of the effort was concentrated on the integrating framework of the environment, that is, those components (user interface, database, executive) which provide for the integration of tools supporting various phases and inter-phase activities of the life cycle.

Payoff:

- Advanced software engineering and development support throughout the entire software development and support life cycle.
- An R&D base for future advances in life cycle software engineering and development technology.

Major FY89 Accomplishments:

- Development of the Software Life Cycle Support Environment (SLCSE) was completed.
- User orientation was held at RADC in August 89.
- Contract was extended to support Beta testing at three (3) AFLC centers during FY90.

Major Users and Related Activities: Beta testing is in progress at Warner-Robins Air Logistics Center, Robins AFB GA.

Installations planned for 2nd quarter FY90 include: Electronic Systems Division, Hanscom AFB MA; Sacramento Air Logistics Center, McClellan AFB CA; Ogden Air Logistics Center, Hill AFB UT.

C.1.8.11 Air Force: Rome Air Development Center

Title: SLCSE Technology Exploitation

Objective: To provide a technology assessment of the RADC Software Life Cycle Support Environment (SLCSE) to fully exploit the existing SLCSE technology and tools and recommend future areas of development.

The overall contribution and benefit of the SLCSE for software development will be analyzed to determine how adequately SLCSE addresses and helps solve the Air Force's critical software problems (e.g., development cost/schedule, product quality, and supportability).

Approach/Thrusts: This effort is being accomplished under a procurement directive to The MITRE Corporation. MITRE will 1) investigate SLCSE capabilities and determine SLCSE's applicability and adequacy to the Air Force software development process, 2) provide a plan to exploit and transition the SLCSE to the Electronic Systems Division System Program Offices and their contractors, and identify any changes required to tailor the SLCSE to Electronic Systems Division requirements, and 3) provide recommendations of emerging technologies and tools (e.g., CASE, natural language, knowledge-based, object-oriented, etc.) which would be appropriate directions for future enhancements of SLCSE.

The final product will be a series of technical reports.

Payoff: A technology assessment of SLCSE to fully exploit the existing SLCSE technology will assure that SLCSE or a SLCSE-like technology can be employed with the greatest possible advantage to all software development efforts DoD wide.

Major FY89 Accomplishments:

- FY90 New Start

Major Users and Related Activities: As a result of this effort, a plan for technology transfer of the SLCSE from the Electronic Systems Division Command Center Evaluation Facility to the System Program Offices and their contractors will be developed.

C.1.8.12 Air Force: Rome Air Development Center/Strategic Defense Initiative

Title: RISC Ada Environment

Objective: Design, implement, test and document a production quality Ada programming environment for radiation-hardened, 32 bit microprocessors.

Tools developed under this effort shall consist of a production quality Ada compiler, a symbolic debugger with simulation capabilities, a macro-assembler and a linker.

Approach/Thrusts: Analyze four RH-32 Phase I instruction sets. Integrate developed environment into RADC Software Life Cycle Support Environment (SLCSE). Develop and present user and maintenance courses. Perform continual error corrections, documentation updates and enhancements.

Payoff: Support the development of mission critical applications for the RH-32 microprocessor in the Ada programming language. Supports the Strategic Defense Initiative (SDI).

Major FY89 Accomplishments: Contract award was 22 Aug 89. Analyzed and evaluated four RH-32 instruction sets for suitability as targets for an Ada compiler and run time system.

Major Users/Related Activities: SDI, WRDC, Space Division/Advanced Tactical Fighter, Boost Phase Surveillance and Tracking Satellite and Space Boost Surveillance and Tracking Satellite.

C.1.8.13 DARPA

Title: DARPA Software Tools Program

Objectives:

- Advanced software environment technology for development and evolution of large-scale systems.
- Persistent object management technology for CAD systems, including software environments.
- Means to integrate formal methods technology into software engineering in order to develop systems with high levels of assurance of correctness for selected requirements.
- Technology to support software and systems prototyping in support of requirements engineering and systems design.
- High-level flexible formal descriptions of systems architectures and interfaces in order to support specification and design reuse, particularly for domain-specific architectures.

Approach/Thrusts:

- Develop environment architecture and interfaces to address development of concurrent, real-time systms using explicit process models to regulate larger scale activities.
- Develop persistent object management ("object-base") technology supporting search, structured types, multiple versions, security and usage metering, and unreliable underlying distributed computing.
- Enable hybrid use of empirical/testing techniques and analytical/formal techniques in systems design, development, and verification.
- Provide tools to make incremental progress from prototypes to production-quality systems on a componentwise basis in heterogeneous software systems.

Payoff

- Open architecture software environment technology that can support full lifecycle needs for concurrent, real-time, and high assurance systems.
- Common data management systems level above operating systems for use in heterogeneous distributed environments.
- Technology means to support early validation and design prototyping for software systems.
- Ability to develop large scale software/hardware systems with very high levels of assurance provided for function, performance, security, safety, and other properties.

Major FY89 Accomplishments

- Implementation of zero-overhead runtime computational checking of certain formal specifications for Ada systems using parallel systems implementations.
- Technique and prototype tools for integrating separately modified versions of the same program, as occurs in programming teams.
- Initial definition of common environment interfaces for measurement, metrics, and empirical analysis tools.
- Fast streaming protocols for heterogeneous distributed systems.
- Protocols for transporting and storing structured objects, including objects with shared substructure, in a network environment. These will be the basis for a prototype object repository implementation.
- Demonstration of initial scalability for formal methods applied to verified program with verified compiler, assembler, and microprocessor definition.
- Formal definition of 32-bit Core RISC (Reduced Instruction Set Computer) architecture.

C.1.8.14 Strategic Defense Initiative/Army: Strategic Defense Command

Title: Distributed Computing Design System (DCDS) -- BMC^3 Experiment

Objective: To provide an integrated software engineering environment that creates/generates Ada code for large distributed, real-time systems in support of SDI programs and experiments and will be available to all SDI government agencies and contractors.

Approach/Thrusts:

- Facilitate technology transfer and establish a STARS technology center
- Maintain operational DCDS code
- Provide DCDS software maintenance
- Perform DCDS configuration management
- Perform acceptance testing
- Provide user training and support
- Modify DCDS per Change Control Board direction
- Document problems and recommend solutions to Change Control Board

Payoff:

- Support to:
  - — SE
  - — EVPA
  - — BM/C^3
  - — NTF/NTB
  - — TES
  - — SDDS
  - — TESSE
  - — Algorithm Architecture

Major FY89 Accomplishments:

- Completed the development of DCDS/Ada for the Sun computer - which is the state-of-the-art for developing large, real-time, distributed Ada systems
- Completed the enhancements on the Sun and VAX versions of DCDS/Ada that provide automatically generated simulations
- Added the automatic generation of 2167A documentation to the DCDS/Ada toolset
- Enhanced the DCDS/Ada methodologies to support the spiral model, a risk driven model of software development
- Added configuration management support to DCDS/Ada
- Completed the development of a 40 hour training course for DCDS/Ada
- Training courses on DCDS/Ada to support users in place and being conducted

Major Users and Related Activities:

- SDS Systems Engineer: Provide simulation capability and access to additional tools
- NTF/NTB: Provide SEE for level 2 simulation coordination center
- TESSE: Provide SEE for TESSE software development
- EVPA: Provide SEE for Experimental Version / 88 (EV88) follow-on
- Algorithm Architecture: Provide SEE for optimally matched BM/C^3 algorithms and computer architecture system
- SDDS: Provide basis for a full life-cycle systems engineering environment (SEE)
- Pilot Command Center builds 1-4: Support integration as a common SEE among SDS elements
- SDS Elements: Available as initial SEE
- Other: Available to all SDI government agencies an contractors as required

C.1.8.15 Strategic Defense Initiative/Army: Strategic Defense Command

Title: Strategic Defense Development System (SDDS)

Objective: To support the production of SDS hardware designs, firmware and software, and to support overall SDS management, design and test through rapid prototyping and simulation.

Approach/Thrusts:

- Develop SDDS core prototype to prove feasibility
- Develop SDDS core system that integrates selected tools and databases

Payoff:

- Support to:
  - — BM/C^3 testability
  - — Algorithm development
  - — Affordability determinations
  - — Software engineering
  - — Security
  - -- Technology identification

Major FY89 Accomplishments:

- Development methodology and architecture were designed
- Demonstrated automatic generation of code from a high-level specification language to be used in identifying requirements
- Completed the definition of the schema and computational model of SDDS that captures the requirements, specifications, design and implementation level information for a system
- Mapped the DCDS schema into the SDDS system Design Language (SSDL)
- Designed and presented requirements for the SDDS common user interface and database
- Designed and documented the Common Intermediate Design Language, the design level component of SSDL
- Completed the design for a translator for mapping DCDS System Specification Language into SDDS/SSDL
- Completed the design for July 1989 demonstration that will map DCDS System specification Language specifications into SDDS/SSDL, support refining the specifications through a graphical editor and generate executable code that implements the specifications

Major Users and Related Activities:

- SDDS is the evolutionary follow-on to DCDS and all DCDS users will have the opportunity to transition to SDDS. Potential users of DCDS are:
  - — System engineer
  - — EVPA
  - — SDS elements
  - — Possible commercial users
  - — NTF/NTB
  - — Algorithm architecture
  - — Government agencies and contractors

C.1.8.16 Ada Joint Program Office

Title: Ada Programming Support Environment (APSE) Evaluation and Validation

Objective: To provide a focal point for addressing the Ada community's need for Evaluation and Validation technology. Assess Ada Programming Support Environments and components.

Approach/Thrusts: Develop methodology for assessing complex Ada Programming Support Environments and their associated tools. Maintain an historical base of data on APSE implementation changes resulting from research and rapid technology advances.

Payoff: The importance of choosing an APSE is manifest when the investment is made to support large (millions of lines of code) critical systems. APSE's represent a major cost to software developers which will affect the software maintenance effort throughout the system life-cycle.

Major FY89 Accomplishments:

- Development and fielding of the initial version of the Ada Compiler Evaluation Capability (ACEC)
- Development of the Evaluation and Validation Reference System, Version 2.0
- Initial development of the Common Ada Programming Support Environment (APSE) Interface Set (CAIS) Implementation Validation Capability

C.1.8.17 Ada Joint Program Office

Title: Ada Language System/Navy (ALS/N)

Objective: The ALS/N programming support environment will provide the capability to use Ada in the current widely installed base of AN/UYK-43, AN/UYK-44, and AN/AYK-14 Navy standard embedded computers. The ALS/N runtime environment provides support and maintenance for fielded applications.

Approach/Thrusts: The Ada Language System/Navy will deliver an integrated collection of program development and maintenance tools. There will also be an underlying database supporting sophisticated configuration management. ALS/N Version 1.6 has been installed at 15 sites, work continuing consists of testing, system enhancements and tools development.

Payoff: In FY88, completed ALS/N retargets for AN/UYK-43 and -44 computers; Ada Compiler Validation Capability for AN/UYK-43 & -44 and AN/AYK-14; began work on the run time environment for multi-processing, multi-programming, and distributed software.

C.1.9 Environment Frameworks

C.1.9.1 Navy: Naval Ocean Systems Center

Title: Software Technology Project: Environments Task

Objective: Establish a reusable framework or infrastructure which supports the low cost, rapid integration of tools and their effective use in Navy software development.

Approach/Thrusts:

- Concept exploration
  - — Development and documentation of Environment Concepts
  - — Evolution towards system specifications of environments/environment components
- Experimentation
  - — Investigations and experimentations of potential building-blocks
  - — Development of prototypes for concept and feasibility demonstration (using building-blocks)
  - — Selected Navy in-house development efforts
- Demonstration
  - — Building-block demonstrations (and transitions)
  - — Prototype demonstrations (and selected applications)

Payoff:

- Supporting environment

Major FY89 Accomplishments:

- Produced first discussion draft of UIS Concepts
- Conducted potential building-blocks investigations/experimentations: (Atherton)

Major Users and Related Activities:

- Cooperation with DARPA/ARCADIA Consortium - Tasks and Technology
- Cooperation with Air Force (RADC) - technology

C.1.9.2 Ada Joint Program Office

Title: Common APSE Interface Set (CAIS)

Objective: Provide a standard interface between APSE tools and operating systems to increase the portability of tools across operating systems.

Approach/Thrusts:

- Develop military standard for CAIS
- Develop prototype implementations of CAIS
- Demonstrate appropriateness of CAIS to specific tools and environments

Payoff: Reduces dependence upon single-vendor for APSE tools, eliminates need to modify tools as hardware and operating systems change, promotes portability of personnel since they can take their tool kit with them from project to project.

Major FY89 Accomplishments:

- Approved revision A of CAIS standard, MIL-STD-1838A, on April 6, 1989; published in October 1989
- Continued prototyping efforts
- Publicized availability of standard
- Agreed to work with Europeans to merge CAIS and the Portable Common Tools Environment+ interface standard.

C.1.9.3 Defense Logistics Agency

Title: Standard Human/Workstation Interface

Objective: Determine DLA standards issues for defining application and workstation functions to present a consistent working environment to all users.

Approach/Thrusts:

- Analyze applications using human-factors engineering techniques to determine general design parameters for interfaces.
- Map application functions to specific keyboard sequences for all DLA workstation types.
- Develop presentation techniques of applications and screen layouts for all DLA workstation types.
- Identify integration tools that allow the user to extract data from one application and input it to another application or local program.

Payoffs:

- Reduce the amount of training required for new users and when introducing new applications.
- Increase use efficiency by reducing errors when switching between applications and simplifying the extraction of data.

C.1.10 Knowledge Based Development Tools

C.1.10.1 Air Force: Wright Research and Development Center/Avionics Laboratory

Title: 2003 Expert Avionics Code Modification

Objectives:

- Reduce the amount of resources (in terms of time, cost, and manpower) required to modify (enhance/maintain) software for processors utilized in avionics applications.
- Apply knowledge Representation & Acquisition methods to develop an expert knowledge base for a specific software package under development.
- Integrate into the RADC SLCSE environment.
- Demonstrate reduction of resources (manpower and money) required to maintain real-time Ada avionics software targeted for MIL-STD-1750A processors.

Approach/Thrusts:

- Intelligent avionics program maintenance tools embedded in an advanced software development/test environment
- Application-specific maintenance capability
- Advanced knowledge acquisition/representation techniques

Payoff:

- Enhanced understanding of Ada code structure
- Better insight into code changes (ripple effects)
- Improved maintainability/code quality

C.1.10.2 Air Force: Rome Air Development Center

Title: Knowledge-Based Development Assistant

Objective: To develop a knowledge-based system that will automate and assist in the transformation of software system specifications to executable code.

Approach: To define the transformational formalism, capture the knowledge and decision making processes used by human programmers, and implement a prototype model demonstrating the assistance that is possible through

automation of this formalism. Artificial intelligence and automatic programming techniques will be used.
Payoff: This tool will show the efficacy of the transformational approach in support of developing and maintaining large, real-time Air Force software projects.
Major FY89 Accomplishments: The preliminary implementation of the Development Assistant has been implemented on the Sun Workstation. Functionality includes the ability to optimize the selection of data structures and algorithms over a limited domain. In addition, an initial version of software transformation history capture and replay has been implemented. The concept of "tactic" has also been developed, allowing the user to work at a more abstract level.
Major Users and Related Activities: This model will be included in the Knowledge Based Software Assistant and made available to the KBSA tech transfer consortium.

C.1.10.3 Air Force: Rome Air Development Center

Title: Knowledge-Based Performance Optimization in Ada
Objective: To develop a tool which will produce efficient, real-time Ada code by refining high level software specifications.
Approach: To use the technology demonstrated in the Performance Assistant and apply it toward automating the design of production quality algorithms and data structures in the Ada language.
Payoff: Optimized and near-optimized data structure and algorithm selection in Ada in support of large, real-time Air Force software projects.
Major Users and Related Activities: This model will be included in the Knowledge Based Software Assistant and made available to the KBSA tech transfer consortium.

C.1.10.4 Air Force: Rome Air Development Center

Title: Annual Knowledge Based Software Assistant (KBSA) Conference
Objectives:

- Provide a forum for discussion of the application of artificial intelligence technology in formalization and automation of the processes of system development and maintenance.
- Facilitate exchange of technology and dissemination of KBSA related research products.
- Encourage participation and competition in the KBSA research program by making the technology and status available to the widest possible community.
- Inform, educate, and involve potential users early in the KBSA program.

Approach/Thrusts: This effort provides the administrative support required to plan, organize, and run an annual conference. The work consists of managing attendance, acquiring facilities, preparing proceedings, organizing mailings and administering all monetary transactions.
The output from this task includes the conference itself, the proceedings, and a final report which includes a summary evaluation of the conference compiled from attendee comments.
Payoff:

- Influx of ideas and technology from a larger community. Efficient technology transfer and independent program evaluation.
- Development of a larger base for competitive procurement.
- Early promotion of KBSA paradigm to potential users.
- Minimization of research duplication.

Major FY89 Accomplishments: 4th annual conference held 12-14 Sep 89.
Major Users and Related Activities: Ultimate users will be government and industrial organizations involved in the development and maintenance of large systems employing software.
Immediate beneficiaries are KBSA Technology Transfer Consortium which receive KBSA program products, formal training, and technical consultation. Members of this consortium include major aerospace corporations, academia, commercial firms, and DoD organizations.
Significant progress in the KBSA program has been achieved using for leverage research results from DARPA, NOSC, and other RADC programs.

C.1.10.5 Air Force: Rome Air Development Center

Title: KBSA Framework
Objective: Identification and specification of support environment capabilities common to and sufficient for all Knowledge Based Software Assistant life cycle components to serve as a unifying basis for future research efforts.
Approach/Thrusts:

- Analyze existing implementations representative of automatic programming systems, knowledge based expert systems, very high level languages, and software engineering environments, identifying supporting technology.
- Determine the support requirements for software life cycle assistant components.
- Specify minimum "common" support environment capabilities.
- Empirically verify requirements and design.

Payoff:

- Reduction in future technology integration efforts.
- Identification of present technology "short falls"
- Reduced duplication of research effort developing support environments.

- Clear definition of support environment issues.

Major FY89 Accomplishments:

- Standard User Interface Design
- Design of a CLOS (Common Lisp Object System) Metaclass for Distributed Objects

Major Users and Related Activities:

- Immediate benefit to KBSA program and contractors through standard support capabilities.
- Framework specification will serve as the "living" specification of the necessary support environment.
- User interface environment specifications are being proposed to the ANS technical committee X3J13 (Lisp).

C.1.10.6 Air Force: Rome Air Development Center

Title: Requirements/Specification Facet for KBSA

Objective: Automated knowledge-based assistance in capturing and analyzing informal user requirements and transforming these requirements into formal executable specifications.

Approach/Thrusts:

- Build upon earlier individual efforts addressing requirements capture and incremental formal specification to provide a seamless acquisition and transformation process from informal user requirements through formal executable specifications
- Define formal transformations from requirements representations to executable specifications.
- Define mechanisms for recreating informal views of requirements from formal specification representations.

Payoff:

- Automation of requirements definition and system design.
- Reusable libraries of domain knowledge.
- Improved traceability, consistency and completeness of requirements and specifications.
- Automatic prototyping and of systems via executable specifications.
- Greater user involvement, early validation of requirements.
- Accurate, automatically generated documentation.
- Support for modification/maintenance at the requirements/specification phase.

Major FY89 Accomplishments: Contract award Jun 89.

Major Users and Related Activities: Ultimately, government and industrial organizations involved in the development and maintenance of large systems employing software.

Immediate beneficiaries are KBSA Technology Transfer Consortium members which receive KBSA program products, formal training, and technical consultation. Members of this consortium include major aerospace corporations, academia, commercial firms, and DoD organizations.

C.1.10.7 Air Force: Rome Air Development Center

Title: Activity Coordination Formalism Design

Objective: Design a visual formalism for representing process models that is abstract and intuitive yet semantically well defined. This formalism shall be useful for describing and coordinating activities and communications such as encountered in large scale software system development, involving multiple, independent, and intelligent agents.

Approach/Thrusts:

- Analyze the communication and coordination protocols underlying activities that occur in actual systems development processes, and identify high level abstractions for expressing these protocols.
- Design a visual protocol formalism that supports the identified abstractions and provides an intuitively accessible notation for specifying activity interaction.
- Develop techniques for deriving detailed enactable process models from the visual protocol formalism.

Payoff:

- Improved project coordination and communication resulting in greater productivity.
- Automation and enforcement of project policy.

Major FY89 Accomplishments:

- Preliminary design of visual protocol formalism.
- Modeling and analysis of Electronic Systems Division system acquisition process.

Major Users and Related Activities: Ultimately, government and industrial organizations automating complex processes such as software developments and system acquisitions.

Immediate beneficiaries are the KBSA program itself and the KBSA Technology Transfer Consortium members which receive KBSA program products, formal training, and technical consultation. Members of this consortium include major aerospace corporations, academia, commercial firms, and DoD organizations.

The research performed under this effort directly complements work being sponsored by the Naval Ocean Systems Center (NOSC) under the Small Business Innovative Research (SBIR) program.

C.1.10.8 Air Force: Rome Air Development Center

Title: KBSA Technology Integration

Objective: Independent critical analysis and evaluation of competing products and technology from the KBSA program with regards to feasibility for consolidation and integration within a practical KBSA environment.

Approach/Thrusts:

- Analyze and evaluate products of the ongoing KBSA program.

- Identify, categorize and enumerate capabilities and technologies identifying maturity and suitability.
- Experimentally evaluate capabilities and product usefulness.

Payoff: Minimization of alternative research pursuits.

Major FY89 Accomplishments: Project initiation 1 Oct 89.

Major Users and Related Activities: Intended to assist RADC in planning future KBSA research program.

C.1.10.9 Air Force: Rome Air Development Center

Title: KBSA Technology Requirements Definition

Objective: To develop a strategy for focussing future KBSA exploratory development in key enabling technologies.

Approach/Thrusts:

- Acknowledged experts in technical areas related to the KBSA will be sponsored to participate in a series of workshops.
- Individual experts will propose strategies for accomplishing goals in specific technical areas.
- Research program strategies will be refined by workshop participants.

Payoff:

- Concentrated research in technologies fundamental to success of KBSA.
- Greater research program effectiveness.

Major Users and Related Activities: RADC and other DoD laboratories, and contractor research organizations addressing the solution of system development problems through the use of artificial intelligence.

C.1.10.10 Air Force: Rome Air Development Center

Title: AI Based PM for Ada Systems

Objective: Increase large-embedded-systems development productivity and lower software life-cycle costs through automated project management.

Approach/Thrusts:

- Piggyback on a preliminary effort that developed the phase-1 project management component (called the Project Management Assistant or PMA) of the Knowledge Based Software Assistant (KBSA). By continuing to use the latest artificial intelligence (AI) technology for automating software project management, this new phase-2 effort (PMA2) will:
  - Expand the "formal knowledge" of software project management developed under phase 1, and further enhance phase-1 PMA capabilities.
  - Provide a mature management mechanism for the knowledge-intensive activities that constitute software development.
  - Retain the reasoning history that goes into software development so that maintenance can be done with greater effectiveness and lower cost.
  - Integrate this advanced project management tool with RADC's Ada-supporting Software Life-Cycle Support Environment (SLCSE)
- Deliver: two advanced PMA prototypes, one for stand-alone use and the other for SLCSE integration; documentation and training.

Payoffs:

- Evolutionary development of KBSA, and availability of a stand-alone product very close to "productization."
- Technology transfer: provide users of the current software development paradigm ("waterfall") with the advanced project management capabilities developed under KBSA.
- Provide a vehicle for valuable experimentation.

Major FY89 Accomplishments: Highly successful demonstration of evolving PMA prototype at the 4th annual KBSA Conference in September.

Major Users and Related Activities:

- Ultimately intended for DoD developers of large, embedded software systems, but PMA can also be used very effectively to manage non-software projects.
- KBSA Technology Transfer Consortium comprised of large-business DoD contractors, small-business DoD contractors, academia, current KBSA developers, and RADC.

C.1.10.11 Air Force: Rome Air Development Center

Title: KBSA Concept Demo

Objective: Clearly demonstrate how the Knowledge Based Software Assistant (KBSA) approach to software development differs from and improves upon the "waterfall" approach.

Approach/Thrusts:

- Develop a "broad" and "shallow" software life-cycle processing system based on the Knowledge Based Software Assistant (KBSA) paradigm: broad in the sense that it will demonstrate support for the entire system life cycle, including both development and evolution; shallow in the sense that it will provide less powerful assistance and functional completeness then will be required in eventual productized versions of KBSA. This development will:
- Design and implement a KBSA system-concepts presentation facility to be integrated with the life-cycle processing system.
- Integrate KBSA technology concepts developed thus far into a single software life-cycle processing system.
- Gain leverage from current industry research in, or related to, KBSA technology areas.

- Deliver: a robust KBSA concept demonstration system with accompanying presentation facility; documentation and training; transferrable use licenses to permit non-DoD sites to exercise and experiment with the system.

Payoffs:

- New insights to guide the direction of phase-2 and phase-3 KBSA contractual efforts
- A robust demonstration vehicle for state-of-the-art KBSA technologies and concepts.
- A technology transfer vehicle.
- Acquisition of essential experience with the KBSA paradigm as a whole.

Major FY89 Accomplishments: Initiation of contractual effort in September.

Major Users and Related Activities:

- Intended for DoD contractors to assist in their eventual transition from current software development and maintenance methods to KBSA technology.
- KBSA Technology Transfer Consortium comprised of large-business DoD contractors, small-business DoD contractors, academia, current KBSA developers, and RADC.

C.1.10.12 Air Force: Rome Air Development Center

Title: Formal Software Specification Verification

Objective: Provide an additional degree of confidence that KBSA-developed software operates in accordance with the intentions of its Air Force designers and users.

Approach/Thrusts:

- Augment KBSA's specification language to provide a mechanism for formally connecting specifications and requirements in a way that will permit a greater degree of verification tuat the specifications do indeed meet the stated requirements.
- Build a prototype implementation of the verification facility intended to demonstrate the viability of the verification concept.

Payoffs: Productized KBSA should offer a means for removing all potential mission-stopping or mission-distorting errors from Air Force Mission Critical Computer Resource software (e.g., applications involved with computer security, flight control and nuclear safety, strategic weapons).

Major FY89 Accomplishments: Preliminary definition of the effort.

Major Users and Related Activities:

- Ultimately intended for DoD contractors. Some potential Air Force users are the Electronics Security Command, Strategic Air Command and Space Command.
- KBSA Technology Transfer Consortium comprised of large-business DoD contractors, small-business DoD contractors, academia, current KBSA developers, and RADC.

C.1.10.13 Air Force: Rome Air Development Center

Title: SLCSE Knowledge-Based Enhancements

Objective: The objective of this effort is to: 1) investigate and determine the potential application of knowledge-based technology and its benefits to the RADC Software Life Cycle Support Environment (SLCSE), and 2) specify potential approaches to augmenting and enhancing SLCSE with an effective knowledge-base and associated knowledge-based tools.

The SLCSE is a computer-based environment which supports the development and post deployment support of mission critical computer systems (MCCS) software in accordance with DOD-STD-2167A.

Approach/Thrusts: Multiple approaches to the knowledge-based enhancements and augmentation of SLCSE will be defined, and as an option, two instances of knowledge-based technology will be developed and demonstrated within the SLCSE.

Knowledge-based technology thrusts within SLCSE include support for 1) its instantiation and maintenance for various types and sizes of software development projects, 2) the future development, integration, and operation of software which will be knowledge-based, 3) the development of software in accordance with the Defense System Software Development Standard (DOD-STD-2167A) life cycle model and applicable development methodologies, and 4) the creation, generation, and revision of documentation in accordance witl DOD-STD-2167A.

Payoff:

- reduction in the level of expertise and manpower to instantiate (i.e., set-up) and manage the environment
- increased maintainability of the environment
- increased user productivity and reliability

Major FY89 Accomplishments: None - effort commenced 19 Sep 89

Major Users and Related Activities: A follow-on to this effort will design and implement a subset of the SLCSE components identified in this effort for knowledge-based technology insertion.

C.1.10.14 Air Force: Rome Air Development Center

Title: Knowledge Based Software Assistant

Objective: Develop engineering tools and methods for design, implementation and integration of knowledge based systems for software engineering.

Approach/Thrusts:

- Use of knowledge-based technology to:
  - — Capture software engineering and applications expertise

  - "Formalize" processes
  - Mediate processes and tasks
  - Share appropriate knowledge between tasks/people
  - "Understand" the system built

Major FY89 Accomplishments: Demonstrated

- Expert systems development tools benchmark
- Various "facets"
  - Program managers assistant feasibility demo
  - Specification assistant
  - Performance assistant

Major Users and Related Activities:

- Major users
  - AFLC (Project Management Assistant for SLCSE)
  - Joint Strategic Planning Staff (JSTPS) (SAPE)
  - USTRANSCOM & MAC (CAMPS)
- Related activities
  - Knowledge-Based Software Assistant (KBSA) Technology Transfer Consortium
  - KBSA Annual Conference
  - RADC AI technology fair

C.1.11 Documentation

C.1.11.1 Army: CECOM

Title: Documentation Engineering (A094 Tactical Software Engineering Technology)

Objective: Investigate technologies, policies and standards supporting the creation, management and use of documentation associated with the software development process.

Approach/Thrusts:

- Review standards and policies to assure that they support good, useful and affordable documentation
- Establish an Industry/Government Task Force on documentation
- Investigate technologies that aid in documentation engineering
- Develop prototypes and demonstrate proof of concept

Payoff:

- Provide demonstration of technologies for documenting large complex systems
- Prototype a documentation engineering system
- Develop a framework for documenting that will work within the current standards and policies
- Better utilization of documentation and reduction of documentation costs

Major FY89 Accomplishments:

- Established Industry/Government Task Force
- Initiated the development of an advanced software documentation demonstration system

C.1.12 Test and Evaluation

C.1.12.1 Air Force: Rome Air Development Center

Title: Ada Test and Verification System (ATVS)

Objective: This effort has developed an integrated set of computer-based tools to provide test and verification support for the Ada programming language (MIL-STD-1815A). The ATVS provides automated assistance during the Code and Unit Test, CSC Integration and Testing, CSCI Testing, and Maintenance phases of the software life cycle.

Approach/Thrusts: The technical approach is to develop test and verification capabilities which include various static and dynamic analyses. In addition to the traditional error detection analyses, the ATVS will provide Ada programming standards enforcement and software quality measurement data collection.

Two version of the ATVS have been developed: 1) a stand-alone version hosted on a Digital Equipment Corporation (DEC) VAX Computer System, and 2) an integrated version which is a component of the RADC Software Life Cycle Support Environment (SLCSE) also under development by RADC. The SLCSE is a computer-based environment which supports the development and post deployment support of mission critical computer systems (MCCS) software in accordance with DOD-STD-2167A.

Payoff: Use of the ATVS will help improve the reliability and maintainability of Ada software systems.

Major FY89 Accomplishments: ATVS software was installed and demonstrated at RADC. A three-day training course was subsequently held at RADC and was attended by both government and industry representatives.

A commercial version of the ATVS has been developed by the contractor (General Research Corporation).

Major Users and Related Activities: Several ATVS beta test sites have been established (e.g. Texas Instruments, Rockwell International, Draper Lab). The results of these test sites will be used to improve the overall quality of the ATVS and help provide direction for future efforts concerning the test and verification of mission critical computer system software.

C.1.12.2 Air Force: Rome Air Development Center

Title: Strategies for Testing Parallel Software

Objective: To develop innovative approaches for testing software for high performance computers.

Approach/Thrusts: To review and identify potential approaches to testing parallel software, assess their potential and conduct proof of concept demonstrations.

Payoff: This effort will contribute recommendations on several promising testing techniques for parallel software.

Major FY89 Accomplishments: Identified the following testing problems: (1) monitoring execution, (2) analyzing traces of execution and (3) generating test cases. Potential solutions include compensating for time delays caused by monitoring, use AI reasoning systems, and to treat the order of execution as an input to the system instead of an output.

Major Users: Any user who will have to test parallel software.

C.1.12.3 Air Force: Rome Air Development Center/Strategic Defense Initiative

Title: Program Mutations for SDI Applications

Objective: To demonstrate the feasibility and effectiveness of software testing based on a technique known as program mutations. Mutation testing is a process where errors or "mutants" are inserted into a computer program. The computer program is then executed with test data. Mutants which are not affected by the test data indicate either weak test data or a program flaw.

Approach/Thrust:

- Develop a prototype mutation testing system for Fortran.
- Investigate a set of program mutants necessary to support Ada.
- Investigate parallel architectures which will significantly increase the performance of a mutation testing system.

Payoff:

- Significant advance in computer program testing technology.
- A testing technology which subsumes several of the state-of-the-practice testing technologies.

Major FY89 Accomplishments: Fortran mutation testing system delivered and demonstrated in August 89.

Major Users and Related Activities: Any software test activity involving Fortran or Ada based systems.

C.1.13 Post Deployment Support

C.1.13.1 Air Force: Wright Research and Development Center/Avionics Laboratory

Title: 3090 Embedded Computer Resources Support Improvement Program

Objective: Insure the availability of adequate cost effective support methods and technologies for routine support, and for war time rapid turnaround of current and next generation avionics software.

Approach/Thrusts:

- ECS Support
  - — Advanced Multi-Purpose Support Environment (AMPSE)
  - — Built-In-Support Function (BISF)
  - — Ada Distributed Systems Evaluation Testbed (ADSET)
- ECS Readiness
  - — Readiness Technologies
  - — RADAR Support Environment (RSE)
  - — Test Facility Working Group (TFWG)
- ECS Software
  - — Modular Embedded Computer Software (MECS)

Payoff:

- Modular, expandable, cost effective support capability
- Highly distributed and integrated capability
- Software change - less than 72 hours

Major FY89 Accomplishments:

- ADSET testbed operational
- Ada models incorporated into AMPSE
- Ada compiler evaluation
- Avionics reconfiguration studies complete
- Initiated test facilities working group
- Reconfigurable aircraft panel simulation
- JIAWG/Ada 9X participation
- Radar support environment study complete
- Support advocacy showing up in new programs

Major Users and Related Activities:

- Users:
  - — WRALC
  - — SMALC
  - — Canada
  - — Warner-Robbins Air Logisitics Center
  - — Oklahoma City Air Logistics Center
  - — Ogden Air Logisitics Center
  - — Integrated Facility for Avionics Systems Test
  - — Air Force Electronics Warfare Evaluation Simulator
  - — WRALC
  - — GWEF
  - — PRIMES
  - — ARSETS

- Related Activities:
  - Fault tolerance survey - RADC
  - SLCSE - RADC
  - Software Technology Support Center - Ogden ALC
  - Reusable software - 62204F STARS
  - Expert Avionic Code modifier - 62204f
  - Ada insertion/issues - System Program Offices ALCs AJPO

C.2 Artificial Intelligence

C.2.0.1 Army: Army Research Office

Title: Science Problems with Military Applications

Objectives:

- Perform basic research in artificial intelligence and computer science.

Approach/Thrust:

- The present focus is on intelligent systems with an emphasis on automated software engineering, and interactive "intelligent" workstations/systems.

Payoff:

- Would provide the foundation on which intelligent interactive software development environment can be developed.

C.2.0.2 Navy: Office of Naval Research

Title: ONR Intelligent Systems Program: Artificial Intelligence

Objective: Understand the automation and extension of human intellectual skills.

Approach/Thrusts:

- Experimental and theoretical basic research
  - Knowledge acquisition
  - Knowledge representation
  - Automated reasoning

Payoff:

- Science
  - Understanding the nature of intelligence
- Technology
  - Advanced aids to human decision making (expert systems, natural language systems,...)

Major FY89 Accomplishments:

- Naval Surface Weapon Center has acquired the Stanford Metalevel Reasoning System, as have over 100 industrial and academic laboratories
- Naval Air Development Center (NADC) has acquired the UMass distributed problem solving architecture
- Navy Research Laboratory (NRL) has acquired SOAR (CMU System for General Intelligence) for study by its machine learning group

C.2.0.3 Air Force: Rome Air Development Center

Title: RADC AI Technology Program

Objective: Establish a research program and foster growth of an AI technology community addressing Air Force Artificial Intelligence (AI) technology needs.

Approach/Thrusts: Technical efforts will be procured as a result of a Broad Agency Announcement. Contractor interaction and open communication within a larger community of AI researchers will be promoted and facilitated with semi-annual technical reviews and other activities.

Payoff:

- AI research supporting current Air Force requirements in Communications, Target Tracking, Sensor Data Fusion, Signal Processing, and Hardware Reliability.
- Broad, long-term targeting of AI research to the Air Force C^3I domain.

Major Users and Related Activities: Specific effects of research under this program will be determined by the specific technical projects procured and cannot be predicted at this time.

This program will have broad relevance to DoD technology requirements and provide support to current RADC technology activities.

C.2.1 Knowledge Based Systems

C.2.1.1 Air Force: Wright Research and Development Center/Avionics Laboratory

Title: Knowledge Representation into Ada Methodology Program

Objective: Determine the feasibility of using evidence flow graphs (EFG) and Petri Nets to develop verification and validation knowledge base techniques for navigation applications.

Approach/Thrusts:

- Develop methodologies to automatically transition the Adaptive Tactical Navigation (ATN) knowledge base into EFGs.
- Perform verification and validation simulations using EFG and Petri Net techniques

- Develop and test methods for the translation of EFG into Ada code

Payoffs:

- Facilitate the use of expert systems by developing verification and validation techniques
- Increase the maintainability of expert systems by revalidating the knowledge base when new information is added
- Develop methodologies for analyzing/improving the real time performance of an AI system
- Develop techniques to automatically convert different knowledge representations into a unified Ada run time environment

Major FY89 Accomplishments:

- Identify knowledge representation for conversion to EFG
- Complete C version AF software

Major Users and Related Activities:

- Major users:
  - Knowledge engineers
  - Real time AI system designers
  - Knowledge base maintenance personnel
- Related activities (programs):
  - Knowledge representation into Ada for parallel processing
  - Artificial intelligence navigation analysis
  - Adaptive tactical navigation
  - Pilot's associate
  - Advanced Geographical Positioning Satellite receiver

C.2.1.2 Air Force: Wright Research and Development Center/Avionics Laboratory

Title: Knowledge Representation into Ada for Parallel Processing Program

Objective: Develop methodologies for transitioning knowledge representations into an Ada based run-time system which is capable of operating on a fault tolerant parallel processor (FTPP).

Approach/Thrusts:

- Modify the FTPP operating system to mimic the Ada activation framework shell so that Ada-based activation framework objects can be executed on the FTPP
- Implement ATN's activation framework objects (Kram Program) on the FTPP
- Validate the integration process by comparing the outputs from the FTPP hosted ATN system with the actual ATN system developed by TASC
- Evaluate performance speedups realizable from parallel processing on the FTPP

Payoffs:

- Demonstration of a complete transition path, from knowledge representations, through verification and validation, to an efficient, fault tolerate run-time environment
- Increased speed and efficiency necessary to run robust intelligent systems under limited real time restraints

Major FY89 Accomplishments:

- Receive C version AF software
- Host C version AF software on FTPP

Major Users and Related Activities:

- Major users:
  - Knowledge engineers
  - Real time AI system designers
  - Knowledge base maintenance personnel
- Related activities (programs):
  - Knowledge representation into Ada methodology
  - Artificial intelligence navigation analysis
  - Adaptive tactical navigation
  - Pilot's associate
  - Advanced Geographical Positioning Satellite receiver

C.2.2 Machine Learning

C.2.2.1 Navy: Naval Research Laboratory (NRL)

Title: Machine Learning: Reactive plan generation

Objectives:

- Develop methods to provide automatic optimization and adaptation of software based on operating experience
- Implement these methods in a "shell" to demonstrate techniques, allow incorporation within systems

Approach/Thrusts:

- Develop and implement advanced means for credit assignment
- Create a learning system environment for reactive plan learning and optimization
- Implement and evaluate on representative classes of military problem

Payoff:

- Major reduction in software development and maintenance costs. (Current DoD software procurement and maintenance exceed $15 billion/year)

- Effective software for use in autonomous vehicles or for control of systems in highly variable environments

Major FY89 Accomplishments:
- Transition from 6.1
- Rule learning model demonstration

C.2.3 AI Development Methods and Tools

C.2.3.1 Strategic Defense Initiative/Army: Strategic Defense Command

Title: Command and Control Decision Aids Test Environment (C^2 Date)

Objective: Access the feasibility and applicability of knowledge-based Decision Aids; develop the technology; and provide transfer to BM/C^3 subsystem and availability to other domains

Approach/Thrusts:
- Build a software development environment (C^{2}Date) to support rapid Decision Aid development & testing
- Build software simulation to represent strategic defense system and threat
- Build prototype Decision Aids using C^{2}Date
- Test and evaluate C^{2}Date and the prototype Decision Aids; and conduct training

Payoff:
- C^{2}Date has rapid prototyping tools for knowledge representation and inference, user interface design, and scenario building
- Tools are applicable to many domains
- One DA transferred to the EV88 program

Major FY89 Accomplishments:
- Basic phase (FY86-FY88) and option year one (FY88-FY89) accomplishments:
  - — New release of C^{2}Date software tools
  - — Prototype DAs developed for attack assessment and SDS system activation
  - — Test and evaluation, training, installation, demonstration, documentation

Major Users and Related Activities:
- Users
  - — SDS System Engineer
  - — U.S. Army Strategic Defense Command contractors
  - — National Test Bed
  - — Other military organizations (e.g., Rome Air Development Center, Air Defense)
- Related Activities
  - — EV88
  - — Future EVPA
  - — Wargames at NTB
  - — Pilot Command Center

C.2.3.2 Air Force: Rome Air Development Center

Title: Large Scale AI Technology Demonstration

Objective: Design, develop, test and demonstrate a laboratory testbed which will serves as a vehicle for the design, analysis, integration, and evaluation of large scale knowledge-based systems in the context of complex, data-rich military problem domains.

Approach/Thrusts: During the first phase the testbed will be implemented consisting of a hardware/software framework in which both knowledge-based and conventional software systems can be integrated and evaluated. During the second phase an initial experimental system will be implemented on the testbed consisting of knowledge-based tactical mission planner, an object-oriented simulator, a route planner expert system and a large relational database.

Payoff: The testbed will provide an experimental approach for interactive design and implementation of robust systems that embody AI technology, require integration of several technical approaches, integration of subsystems and scaling up of the technology in dynamic, complex and time-constrained military environments.

Major User and Related Activities:
- System designers for large scale and/or real-time AI system applications
- Service Laboratories designing large scale AI technology demonstrations
- SAC, TAC, AFLC
- USTRANSCOM AI Transportation Planning Systems

C.2.3.3 Air Force: Rome Air Development Center

Title: Validation of Artificial Intelligence Software: Phase II SBIR

Objective: Develop specific verification and validation methods, procedures, and tools for AI software. The contribution will be tools and methods used for system acquisitions within the Air Force.

Approach/Thrusts: 1) Categorize types of AI software; 2) Identify potential verification and validation sources; 3) Select verification and validation activities and analyze their applicability to AI software.

Payoff: Standardization of methodologies for the creation of software systems involving Artificial Intelligence techniques. Improved reliability of next-generation software systems involving Artificial Intelligence techniques.

Major Users and Related Activities: AI System Developers

C.2.3.4 Air Force: Rome Air Development Center

Title: Engineering of Artificial Intelligence Software

Objective: Develop specific verification and validation methods, procedures, and tools for AI software. The contribution will be tools and methods used for system acquisitions within the Air Force.

Approach/Thrusts: Building upon extensive previous survey and analysis of quality measures and assurance for AI software, this effort is attempting enhancement and practical demonstration of software engineering techniques such as explicit modeling, string typing, formal semantics, and rule coupling/partitioning algorithms, etc. through rigorous evaluation in practice.

Payoff: New methodologies for the verification and validation of software systems involving Artificial Intelligence techniques. Improved reliability of next-generation software systems involving Artificial Intelligence techniques.

Major Users and Related Activities: AI System Developers

C.2.3.5 Air Force: Rome Air Development Center

Title: Benchmark Investigation/Identification

Objective: Develop a "descriptive terminology", a collection of non-subjective terms with associated definitions and procedures for appropriately applying them, with which software components for processing Natural Language (NL) can be described in a standard way.

Approach/Thrusts:

- Survey open literature for current NL interface terminology and evaluation criteria.
- Develop non-subjective terminology for NL interface input and output processing capabilities, classification system for capabilities, and clear and concise means for applying the terminology.
- Evaluate project developments at six month intervals.

Payoff:

- First step towards objective evaluation capabilities for man-machine interface systems.
- Objective descriptions of system capabilities will allow improved selection of software interface components matched against domain requirements.

Major Users and Related Activities: This effort is necessary for the continued improvement of interface technology. All interface software developers and interface users, including those in the DoD, are in need of a means to describe interface capabilities and requirements in order to direct further research and appropriately apply current and future interface components.

C.2.3.6 Air Force: Rome Air Development Center/DARPA

Title: Validation of Knowledge Based Systems

Objective: Develop specific verification and validation methods, procedures, and tools for AI software. The contribution will be tools and methods used for system acquisitions within the Air Force.

Approach/Thrusts: The development of a software system designed to detect stereotypical errors, questionable constructs and inconsistencies in rule-based systems. The bulk of the concepts embodied have been proven, the majority of the work lies in optimization of the system and in interface construction.

Payoff: A tool which aids in the verification and validation of software systems involving rule-based techniques by detecting stereotypical errors, questionable practices and inconsistencies in rule bases.

Major Users and Related Activities: AI System Developers

C.2.3.7 Air Force: Rome Air Development Center/DARPA

Title: Reasoning With Incomplete Uncertain Information

Objective: The objective of this effort is to develop expert systems technology for reasoning with incomplete and uncertain information within an abstract battle management domain, enabling such systems to: give a reasoned account of the current state of affairs; explain an ongoing reasoning process; alert the decision maker to an incipient problem; generate potential responses to the problem in the form of decision options; make predictions of likely consequences to the various responses; evaluate the response options; and execute the preferred option, monitoring its execution.

Approach/Thrusts: The approach to this technology development is to develop theoretical foundations and representation techniques for handling incomplete and uncertain information, and to validate these theoretical foundations through the implementation of an experimental prototype battle management system.

Payoff: The payoff will be expert system technology that is capable of reasoning with incomplete and uncertain information, that can integrate fragmentary data from various sources, and that can explain its reasoning processes. These capabilities are developed to meet the needs of large military systems.

Major FY89 Accomplishments:

- PRIMO (Plausible ReasonIng MOdule) was designed and implemented.
- Reasoning with Uncertainty Module rules were translated to PRIMO. The translated rules were used as a testbed for debugging and timing of PRIMO.
- Technology was transferred to DARPA's programs: Pilot's Associate, Submarine Operational Automation System, and Air Land Battle Management (ALBM) Situation Assessment Module.

Major users Related Activities: Technology transitioned to DARPA's Strategic Computing Program - Pilot's Associate.

C.2.4 AI Languages

C.2.4.1 Air Force: Rome Air Development Center

Title: Fifth Generation Logic Programming Architectures

Objective: The objective of this effort is to develop an implementation of SUPER, a fifth generation logic programming language.

Approach/Thrusts: It will be accomplished in three phases: 1) design and formalize the Super Abstract Machine (SAM), 2) implement the SAM interpreter/compiler on a standard architecture, 3) test the system by implementing a real-world application.

Payoff: Make new fifth generation languages more available and accessible to the AI user community. Frequently, high-level fifth generation languages are restricted to highly specialized hardware that is not accessible to the community as a whole. This effort will provide a new implementation of a fifth generation logic programming language,SUPER, that is robust yet efficient and available on a variety of conventional and widely available hardware.

Major FY89 Accomplishments:

- Completed the design of the SAM
- Completed a Scheme implementation of the SAM
- Completed a beta version compiler of the D language

Major Users and Related Activities: AI Software developers

C.2.5 AI for Autonomous/Smart Weapons

C.2.5.1 Navy: Naval Research Laboratory

Title: Sensor-based Systems: Control architectures for autonomous vehicles

Objectives:

- Development of a robust architecture for control of autonomous vehicles
- Development of a task behavior language for vehicle mission descriptions
- Demonstration for an interactive prototyping environment for autonomous vehicle control concepts

Approach/Thrusts:

- Design and implement controls tested
- Model existing vehicle as validation platform
- Implement subsumption controller
- Conduct validation tests, iterate design
- Develop behavior language "compiler"
- Extend evaluation to additional platforms

Payoff:

- A key limitation in design of autonomous vehicles is the control architecture
- Development of automated methods for generation and validation of controls is the key to feasibility for most mid-term vehicles

Major FY89 Accomplishments:

- Current activity:
  - — Simulation environment developed, Brooks land vehicle implemented and evaluation
  - — Real time aircraft hierarchical control system modeled, successful demonstration of concept
  - — Hierarchical control system "compiler" implemented

Major Users and Related Activities:

- Three opportunities identified:
  - — LAURA project in Code 5700
  - — SEA LION project in Code 5100
  - — Robotic flight deck vehicle

C.2.5.2 Navy: Naval Weapons Center

Title: Missile Computer Software

Objective: Investigate AI techniques in missile computers

- AI viewpoint: Best fit, Ada
- Measure embedded system performance: Adaptive control, Target selection

Approach/Thrusts:

- Demonstrate System: Simulations
- Weapon System Information - Heuristics: Corporate knowledge, Empirical data
- System design: Fit architecture to problem

Payoff:

- Enhanced missile performance: Adaptive control, Target selection
- Improved software: Design, Maintainability, Reliability

Major Users and Related Activities:

- Major users
  - — HARPOON
  - — High-speed Anti-Radiation Missle (HARM)
- Related activities
  - — Joint Integrated Avionics Working Group: Avionics/Operational Flight Program Requirements

- — Navy Research Laboratory: Genetic Algorithms & Adaptive Systems
- — Ada Joint Program Office: Run Time Performance Criteria
- — Common Ada Missile Packages: Air Force
- — Autonomous Guided Weapon
- — Avionics Program Expert: Air Force
- — Lockheed Missiles/Space: Software Engineering Tools
- — McDonnell Douglas: AI Support Using Ada
- — Rockwell: Symbolic Computing Technology
- — Science Applications International: Ada inferencing system tool

C.2.5.3 Air Force: Wright Research and Development Center/Avionics Laboratory

Title: Adaptive Tactical Navigation

Objective: To develop an expert system that continuously and automatically manages a suite of navigation sensors, taking into account required weapon system accuracy, threat exposure, jamming environment, and navigation system component failure and degradation.

Approach/Thrusts:

- Perform knowledge acquisition with current aircrews and avionics engineers
- Design two candidate architectures well suited for the domain knowledge which satisfy real time constraints
- Develop the system utilizing the most appropriate architecture
- Test the system using a realistic, man-in-the-loop simulation to evaluate the utility and functionality of the system
- Analyze interface/communication requirements

Payoffs:

- Provide navigation information sufficient to meet future tactical aircraft requirements by making the bet use of available navigation resources in the context of the mission
- Improve the probability of survival and target kill under varying threats and mission demands
- Improve crew performance and situational awareness
- Provide an enhanced navigation system failure diagnostic capability and stored diagnostic information for analysis at the conclusion of the mission

Major FY89 Accomplishments:

- FCA completed
- Laboratory demonstration
- Cockpit FCA

Major Users and Related Activities:

- Major users:
  - — Product divisions (Aeronautical Systems Division)
  - — Major airframers
  - — Real time AI system designers
- Related activities (programs):
  - — Pilot's associate
  - — Knowledge representation into Ada methodology
  - — Knowledge representation into Ada for parallel processing
  - — Artificial intelligence navigation analysis

C.2.6 Decision Aids

C.2.6.1 Army: Ballistics Research Laboratory

Title: Air—Land Battle Management (ALBM) (7149)

Objective: Develop Battlefield Decision Aids.

Approach/Thrust:

- Demonstrate three cooperating Expert Systems called FORCES, which are Maneuver at Corps, Maneuver at Division, and Fire Support at Corps.
- Demonstrate the applicability of the Knowledge Based Expert System building environment tool, called STAR.
- Investigate Natural Language message fusion.
- Use ALBM test bed to evaluate decision aids.

Payoff:

- Would provide the technology required for developing Battlefield expert systems.

Major FY89 Accomplishments:

- Initiate validation and verification program of large AI systems using ALBM as a test bed.
- Successful ALBM demonstration of FORCES, March 1989."

C.2.6.2 Army: CECOM

Title: Tactical Automation

Objective: Develop and Demonstrate Transitionability of Expert System Products for

- force level division maneuver planning
- force level division fires planning
- force level C^2 execution monitoring and replanning

- force level electronic warfare planning
- lower echelon tactical maneuver planning
- lower echelon C^2 execution monitoring and replanning
- automation of combat information correlation and reconciliation

Approach/Thrust:

- Develop and demonstrate transition products from the joint AMC/DARPA funded Air—Land Battle Management (ALBM) AI technology development effort.
- Demonstrate form, function, and feel decision support expert system planning and execution monitoring products for insertion into Army Tactical Command and Control System (ATCCS) Force Level Control System.
- Implement variants of ALBM C^2 decision support products specifically scoped and tailored for lower echelon and heavy force modernization C^2 applications

Payoff:

- Significant reduction in the time required to develop Corp, Division, and Brigade level maneuver plans and strategies (six to one reduction ratio projected)
- more time available for lower echelon forces to develop their plans
- more time to develop and investigate alternative tactical maneuver options
- automated expert system support for operations plan execution monitoring and replanning functions

Major FY89 Accomplishments:

- Provided CECOM portion of funding for joint AMC/DARPA ALBM technology development effort
- Major ALBM technology demonstration of moves and fires expert planning subsystems.
- Participation in Heavy Force Modernization Best Technical Approach, demo definition, lower echelon C^2 requirements definition.

Major Users and Related Activities:

- Development of ALBM force level decision support products are being fully coordinated with CAC and Council of Colonels.
- All ALBM transition products will be developed with participation of CAC and other army user designated subject matter experts.
- All ALBM transition products will be evaluated by running it on ATCCS equipment at the Ft. Leavenworth CAC Technology Assessment Center Future Battle Lab and the ATCCS Experimental Station.
- Currently coordinating and working PEO ATCCS, Operational Tactical Data Systems and Advanced Field Artillery Tactical Data Systems personnel to synchronize development of transition product capabilities and availability within their funding, insertion and requirements windows.
- Lower echelon C^2 Decision Support efforts included as essential elements of Heavy Force Modernization Best Technical Approach.
- Lower echelon C^2 Decision Support efforts also support multi-mission area sensor demo effort.
- Support products are being evaluated for use on SIMNET.

C.2.6.3 Navy: Naval Research Laboratory

Title: Intelligent Decision Aids: Bayesian Reasoning Tool (BaRT)

Objective: Develop artificial intelligence methods for effectively dealing with uncertainty in computer-based decision aids. Provide a reusable tool for embedding within decision aid applications.

Approach/Thrusts:

- Develop a generic classification "shell" for efficiently processing classification networks handling uncertain data
- Incorporate within the shell development, auditing and maintenance features to create a BaRT
- Implement and demonstrate parallel BaRT
- Develop run-time BaRT for incorporation in decision support systems

Payoff:

- Efficient implementation of decision aids incorporating evidential reasoning and data fusion
- Operator workload reduction in complex, high workload tasks

Major FY89 Accomplishments:

- System extensions to include influence diagrams, provide capability for computing maximum expected utility
- Interface implementation, system test
- Parallelization demonstration and performance evaluation

Major Users and Related Activities:

- The Bayesian reasoning tool is incorporated in the NRL Decision Support Shell (DSS) and has been supplied to Gould and Mitre for application to classification problems and to three divisions within NRL
- Parallel BaRT is part of the NRL SDI Battle Management Testbed

C.2.6.4 Navy: Naval Research Laboratory

Title: Command Decision Support

Objective: Create a Command Decision Support Shell.

C.2.6.5 Air Force: Rome Air Development Center

Title: CAMPS Portability Enhancements

Objective: Development of a portable, well documented software tool for the Core of A Mission Planning System (CAMPS), an AI planner for stereotypical resource allocation planning tasks.

Approach/Thrusts: Productization of the current CAMPS planner so that it runs in Common LISP and Common LISP Object System, is fully documented, and is readily usable as a tool.
Payoff: Early insertion of advanced planning capability into Air Force Operational planning domains.
Major Users and Related Activities:

- JSTPS/SAPE Program
- TAC
- USTRANSCOM

C.2.6.6 Air Force: Rome Air Development Center

Title: Multi-Level MAC Planning
Objective: Development of the current CAMPS planner into a multi-level planning system capable of supporting near real-time mission planning for a MAC transportation planning domain.
Approach/Thrusts: Application and enhancement of CAMPS planning shell and job shop scheduling techniques to address MAC planning domain.
Payoff: Advanced technology demonstration addressing specific requirements of large scale transportation planning domain.
Major User and Related Activities:

- MAC
- USTRANSCOM

C.2.6.7 Air Force: Rome Air Development Center

Title: Non-Linear Planning
Objective: Research to enhance the application of general AI planning techniques to support knowledge-based reasoning in Air Force domains where pre-existing "expertise" is lacking. Such domains might include new equipment diagnosis, resource management, and tactics development.
Approach/Thrusts: Explore research issues associated with the amount and content of communication required to support planning and replanning in an environment in which there is a centralized strategic planning and a remote tactical planner and plan execution agent.
Specific issues being addressed include: the role of dependencies for planning and execution, communication between execution agent and central planner, the representation of contingencies within disjunctive plans, and complementing planning ad machine learning techniques.
Payoff: The application of knowledge-based systems to domains lacking a priori "expertise" has tremendous potential payoff for the Air Force since it is this lack of "expertise" (identifiable, measurable, and consistently superior human performance models) that currently hinders the widest application of the technology today.
Major FY89 Accomplishments: An initial high level architecture design has been developed characterized by an enhanced task formalism, a means to express an agent's capabilities, an improved plan state definition, and the use of an agenda to provide dynamic control for the activation of activities as well as a method to identify and manage choice points in planning.
Major Users and Related Activities:

- This is a joint RADC/EOARD research project.
- AI applications where a priori "expertise" limits the applicability of state-of-the-art AI technology.

C.2.6.8 Air Force: Rome Air Development Center

Title: Intelligent Real-Time Problem Solving
Objective: Research into the relationships between problem solving strategies and the active, dynamic management of scarce problem solving resources, including time-to-solution resources, as they effect the accuracy, timeliness, and completeness of the problem solution.
Approach/Thrusts: Within a selected problem solving domain (e.g., diagnosis or planning), development and characterization of a broad spectrum of problem solving strategies that are responsive to varying problem solving resource constraints.
Payoff: In conjunction with real-time processing, intelligent real-time problem solving can make computer based decision support systems responsive to time critical Air Force needs. This research will focus early results on Air Force requirements for hard, real-time response from AI based systems.
Major FY89 Accomplishments: Development and documentation of a concise definition of terms and issues that serve to define a domain of inquiry for remaining phases developing techniques to solve common intelligent real-time problem solving issues.
Major Users and Related Activities: This is a joint program between RADC, Air Force Office of Scientific Research and WRDC since requirements for this technology are pervasive across the applications of AI within the missions of RADC and WRDC.

C.2.6.9 Air Force: Rome Air Development Center

Title: AMPS Intelligent Multimedia Interface (AIMI)
Objective: To establish a technology baseline in intelligent, multi-media interfaces to complex knowledge-based systems.
Approach/Thrusts: This effort will enhance an existing intelligent man-machine architecture (KONG) and apply it in the development of a cooperative problem solving interface (AIMI) for the advanced knowledge-based mission planner AMPS. This will provide user-oriented natural problem solving windows into complex AMPS planning capabilities.

Key thrusts include the design, implementation, and demonstration of advanced multi-media user interface technology.

Payoff:

- Short-term: Advanced multi-media user interface technology (including integrated graphics and natural language as well as multi-media presentation strategies) that is domain and task independent (i.e., can be used in interfaces to a variety of knowledge based system tasks such as planning, diagnosis, configuration and so on).
- Long-term: Enhanced productivity and effectiveness of military planning through the development of more natural user interfaces to complex knowledge based systems. A byproduct of this is increased learnability by minimizing training required to utilize advanced automated planning systems (e.g., AMPS).

Major FY89 Accomplishments:

- Design of key AIMI system components including window architecture and actual AMPS interface.
- Common Window protocols from Intellicorp integrated.
- Discussions with TAC planners from Shaw AFB confirm operational preference of AIMI interface (including map interface).
- After extensive survey of map systems, ARC/INFO by Environmental Systems Research Institute is selected. ARC/INFO stores data in a vectorized form in a relational data base. MITRE's Cartographic Services in Dept 71 will assist in acquiring European geographic data at no cost to Air Force.

Major Users and Related Activities: This project is targeted at the next generation interface technology for use in battle management systems, including mission planners.

Close coordination with TAC planners at Shaw AFB provides critical feedback early in the design of AIMI and assures user-driven research objectives.

Commercial and academic developments in interface technology are being closely monitored through participation at and contribution to industrial and professional symposia.

C.2.6.10 Air Force: Rome Air Development Center/DARPA

Title: Survivable Adaptive Planning Experiment

Objective: Develop and demonstrate: (1) a decentralized , distributed data processing system serving geographically separate, though functionally integrated command and control nodes, and (2) develop and demonstrate a high level suite of AI tools and applications.

Approach/Thrusts: From a base of present operating systems (CRONUS, MACH), expand and evolve into a decentralized operating system and distributed database management system. Examine applications for operation in the trans- and post-attack timeframe.

Payoff: Technology base to support the development of a survivable distributed war planning system.

Major FY 89 Accomplishments: Development and demonstration of adaptive planning strategies supporting Joint Strategic Target Planning Staff strategic planning operations.

Major Users and Related Activities: JSTPS

C.2.6.11 Air Force: Rome Air Development Center/DARPA

Title: Constraint Directed Planning

Objective: To develop a general theory of constraint directed reasoning and examine its utility in planning and scheduling.

Approach/Thrust:

- Deduce from the inherent constraints of a planning problem the topological structure of the problem space.
- Exploitation of the topological structure in the efficient search for solutions.

Payoff:

- Provide enhanced general planning and scheduling techniques primarily in a manufacturing domain, under DARPA's Strategic Computing Program.
- Transfer of successful techniques to aerospace manufacturing and Air Force C^3I resource allocation tasks.

Major Users and Related Activities:

- Air Force Large Scale Transportation Planning System Designers.
- Aerospace manufacturing applications under the Strategic Computing Program

C.2.6.12 Air Force: Rome Air Development Center/DARPA

Title: Truth Maintenance in Automatic Planning

Objective: Develop and demonstrate AI planning technology that enables knowledge based systems to effectively plan in the presence of incomplete and uncertain information and to dynamically replan in respond to changes in the environment.

Approach/Thrusts:

- Implementation of a software architecture for non-linear planing that embodies recent advances in truth maintenance technology for non-monotonic reasoning.
- Demonstration of the new planning technology in a NASA or an Air Force C^2I domain.

Payoff: Improve performance by reducing time required for planners to intelligently select goals i the face of uncertainty and non monotonic reasoning.

Major FY89 Accomplishments:

- Completion of an initial prototype capable of accepting initial facts and goals and producing a plan by representing decision dependencies in a truth maintenance network.

- Development of an intelligent strategy to reduce the search time required to resolve conflicts in a partially ordered plan.

Major Users and Related Activities:

- Exploitation of OPUS, a knowledge-based development environment, developed under the DARPA Strategic Computing Program.
- Using Commands with complex, dynamic planning applications involving reasoning with uncertainty and incomplete information - SAC, TAC, NASA.

C.2.7 AI Assisted Logistics and Maintenance

C.2.7.1 Navy: Naval Research Laboratory

Title: Expert Systems for Electronic Maintenance: Fault Isolation Shell

Objectives:

- Develop knowledge-based system techniques to streamline the fault modeling and diagnostic process
- Implement these methods in a "shell" to demonstrate techniques, allow incorporation within systems

Approach/Thrusts:

- Develop causal reasoning model for avionics systems
- Demonstrate feasibility of using causal model-based diagnostic procedure generation for ATE and man-aiding applications
- Implement a fault isolation shell

Payoff:

- Major reduction in ATE software development costs. Current costs of over $700M/year on development could be reduced by 25-50%.
- Improved level of diagnosis expected, reduction of 25% in total number of tests required to diagnose.

Major FY89 Accomplishments:

- Completion of test programmers interface
- Incorporation of feedback from test sites
- Modelling of the NORDA sonar system
- Design of advance user interface
- Documentation completion and distribution

Major Users and Related Activities:

- 9 DoD sites, 6 in industrial sites TIGR project

C.2.7.2 Air Force: Rome Air Development Center

Title: Assurance Technology for Electronics

Objective: Development and application of AI based tools and techniques in support of integrated design for reliability, availability and supportability concurrent with development of system performance capabilities.

Approach/Thrusts:

- Review AI developments for application to R/M/T domain
- Develop tools for incorporating R/M/T at early stages of system development
- Utilize information from electronics theory and failure modes
- Increase efficiency of diagnostics and degree of on-board fault detection and isolation

Payoff:

- Increased system availability
- Reduced repair times
- Reduced maintenance costs
- Reduced logistics support requirements

Major FY89 Accomplishments:

- Completion of MIL-STD tailoring expert system
- Smart BIT/Stress measurement integration initiated
- Completion of maintenance expert system investigation

Major Users and Related Activities:

- AFSC
- AFLC
- Time Stress Management Device - RADC

C.2.7.3 OASD(P&L)

Title: Acquisition - Logistics Artificial Intelligence

Objective: Integrate new data processing technologies into the acquisition and logistics operations to support DoD process improvement objectives of the Defense Management Review.

Approach/Thrusts:

- Mature technologies (e.g. expert systems):
  - Use of AI and other unconventional computing technologies to increase expert knowledge and effectiveness in use of automated data at base level to handle logistics related problems such as inventory, maintenance, transportation, security, safety, and environmental problems;
  - Communication of information on advances in AI and techniques for application to solve military logistics problems.

- Maturing technologies:
  - Develop concepts for using neural computing, parallel computing, and other techniques in the environment of DoD production and logistics systems;
  - Develop and evaluate several prototype applications;
- Other research activities:
  - Cooperation with DARPA to use DARPA sponsored ADP technology developments in acquisition and logistics processes.
  - Identification of production and logistics problems needing new technologies or directions of research.

Payoff:

- Reduced cost of operations
- Increased productivity and effectiveness -- cost avoidance
- Safer working and operating environment
- Improved quality in organization outputs

Major FY89 Accomplishments:

- Development of five year AI logistics insertion plan
- Development of a Service-wide conference on AI and Total Quality Management (Scheduled for March 90).
- Support of Service wide Expert Systems. Neural Nets, and Voice Recognition/expert systems development and implementation.

C.2.7.4 Defense Logistics Agency

Title: Artificial Intelligence System for Cataloging Applications

Objective: Investigate use of AI to automate the parts cataloging process by extracting key cataloging information from descriptive narratives.

Approach/Thrusts:

- Analyze the existing cataloging process and the planned automation schemes to develop a model of the decision making process required in classifying and cataloging supply items.
- Apply expert systems, natural language interfaces, and other artificial intelligence techniques to assist catalogers in their daily transactions.
- Document the results of the demonstration and make recommendations for possible deployment at DLA sites and with designated CALS contractors.

Payoffs:

- Increase the efficiency of the DLA cataloging function while improving the quality of item descriptions produced.
- Allow DLA to retain valuable expertise and train new personnel.
- Prepare the agency for the increased volume of cataloging and complexity of description that will accompany DoD modernization and the CALS initiative.

C.2.7.5 Defense Logistics Agency

Title: Retrieval and Analysis of LSAR Data for Provisioning

Objective: As part of the DRAMA project, DLA is developing a demonstration system to access the standard Logistics Support Analysis Records (LSAR) data, identifying pertinent cataloging information, generating update transaction for the Standard Automated Material Management Systems.

Approach/Thrusts:

- Coordinate with other DLA cataloging projects and the main DRAMA development project at DARPA and ISI.
- Use data on parts for the Air Force C-17 from MacDonnell Douglas.
- Develop a single workstation capable of supporting all cataloging functions and interfacing with the existing DLA information systems.
- Investigate the use of Artificial Intelligence techniques to scan the Supply Support Requests, identify the data needed to assign National Stock Numbers, and generate transactions.
- Focus on demonstrating three capabilities; item entry, requirements determination, and data monitoring.
- Provide data serving capabilities for a group of microcomputers on a Local Area Network or vial a dial-up connection.

Payoffs:

- Support DLA preparedness by providing more timely information on stock levels, engineering changes, and implementations dates.
- Help avoid the purchase of deleted items, identify items for other sources of supply, and reduce out-of-area shipping costs.
- Increase efficiency of DLA personnel by reducing the manual effort required to process Supply Support Requests and establish new National Stock Numbers.
- Assist DLA in developing an integrated Product Manager environment.

C.2.8 AI for Sensor Processing

C.2.8.1 Air Force: Wright Research and Development Center

Title: 2003 Adaptive Network Avionics Resource Manager

Objective: Develop advanced electronic counter measures resource management techniques for use in future combat aircraft.

Approach/Thrusts:

- Develop techniques, primarily AI techniques, to automatically control the on-board electronic counter measures suite
- Compare performance and adaptability of various AI and non-AI techniques

Payoff:

- Increased survivability and adaptability for next-generation aircraft against the dense, dynamic threat environments of the future

C.2.8.2 Air Force Systems Command

Title: Adaptive Network Sensor Processor

Objective:

- Develop prototype sensor signal processor which
  - — Recognizes and classifies familiar signals
  - — Generalizes to classify war mode changes
  - — Analyzes and characterizes novel signals

Approach/Thrusts:

- Two architectures developed and tested in parallel
  - — Professor Grossberg (Boston University)
  - — Professor Anderson (Brown University)
- Test with realistic data
- Implement with parallel processors

C.2.8.3 Air Force: Rome Air Development Center

Title: Sensor Exploitation

Objective: Develop, evaluate, and demonstrate AI technologies to improve and automate the exploitation of sensor data.

Approach/Thrusts:

- Geopositioning, target identification, terrain analysis and predictive battlefield intelligence using model based vision and neural networks

Payoff:

- Time processing of intelligence to:
  - — Determine threats
  - — Identify and precisely locate mobile targets
  - — Identify enemy activities and determine future course of action
  - — Provide warning

Major *FY89* *Accomplishments*:

- Completed version one of the intelligence analyst systems able to monitor and predict missions of foreign space launches
- Developed AI based architecture for automatically recognizing and tactical communications intelligence (COMINT) signals
- Developed a denial and deception planner - provides knowledge based capability for analysis of camouflaged/concealed targets
- Demonstrated feasibility of predictive intelligence at Allied Tactical Operations Centre/SEMBACH

Major Users and Related Activities:

- Major users
  - — AFSPACECOM
  - — ESC
  - — DoD Intelligence Information System
  - — AF INTEL Community
  - — HQ SAC
  - — NSA
  - — HQ TAC
- Related activities
  - — WRDC
  - — Air Force Human Resources Lab
  - — Advanced Armament Materials Research Lab
  - — Electronic Systems Division

C.2.8.4 Strategic Defense Initiative/Army: Strategic Defense Command

Title: Knowledge Based Sensor Data Fusion Program

Objectives:

- Develop adaptive real-time multi-sensor data fusion and battle planning functions through algorithmic and knowledge based evidential methods.
- Demonstrate employment of parallel super microprocessor network.
- Demonstrate variable uncertainty calculi for SDI problems.

Approach/Thrusts: Through the implementation of four cooperating systems perform basic situation assessment, correlation, discrimination, and multi-sensor fusion, provide adaptive real-time response to a changing threat environment. Underlying concepts:

- Parallel microprocessors and other special purpose hardware as required.
- Hybrid system (knowledge base-algorithmic),

- Uncertainty management using variable calculi.
- Accumulate knowledge through machine adaptation.

Payoff

- Adaptive discrimination approach to reactive threats
- Support to algorithm architectures
- Technology demonstration of a unique combination

Major FY89 Accomplishments:

- Established variable uncertainty facilities in operational shell

Major Users and Related Activities:

- Algorithmic architectures: provide hybrid system for matching algorithms to knowledge.
- Demonstration/Validation: integrated into mid-course battle manager.
- Systems Engineer: provides means of evaluating the relative worth of items of information.
- Intelligence: provides automated approach to combining evidence of various types and certainties.

C.3 Distributed/Parallel Processing

C.3.0.1 DARPA

Title: DARPA Distributed Systems Software Program

Objective: Systems software base for heterogeneous distributed and parallel systems. Systems software includes operating systems and database support. Heterogeneous systems include high performance systems, distributed C^3 systems, and network configurations of workstations and servers.

Approach/Thrusts:

- Portable distributed secure operating systems.
- Distributed database paradigms and integrity.
- New operating systems paradigms for parallel computing.

Payoff

- Portable open-architecture operating systems support for parallel and distributed computing for high performance, C^3, and workstation/server networks.
- Portable secure distributed operating systems technology.
- Distributed database systems supporting real-time with unreliable communications.

Major FY89 Accomplishments

- Development of MACH operating system that is compatible with Unix, that enables access to heterogeneous parallel and distributed computing, and that can be made highly secure.
- Transition of MACH to multiple commercial platforms, including workstations and high performance parallel systems.
- High performance Common Lisp for MACH and multiprocessor systems.

Major Users and Related Activities

- MACH is widely available on commercial platforms including Encore, NeXT, BBN, Sun, and others. It is widely used among users of parallel and distributed systems.

C.3.1 Distributed/Parallel System Design

C.3.1.1 Army: AIRMICS

Title: DY10-01/Distributed Systems

Objective: Develop tools, techniques, and prototypes for the design and development of computer systems with architectures that are truly distributed. The computing functions are dispersed among several physical computing elements. These elements may be collocated or geographically separated.

Approach/Thrusts:

- Mathematical models, simulations, and prototypes
- Performance modeling tools
- Static and dynamic reconfiguration
- Load balancing method
- Standards for system evaluation
- Adaptable distributed system

Payoff:

- Cost effective insertion of distributed systems technology
- Shared software and hardware resources
- Processing power closer to user
- Multiple transactions and interprocessing
- Data synchronization
- Data network reliability

Major FY89 Accomplishments:

- Began installation of system at AIRMICS
- Published two research papers
- Tech Assessment completed
- Completed functional description of Requirements and Specification Language

- Defined high performance distributed system modeling environment (SBIR)

C.3.1.2 Navy: Naval Ocean Systems Center

Title: Heterogeneous Computing
Objectives:

- Provide a 6.2 focus for the next generation computer resources (NGCR) program
- Stimulate university research, national initiatives and corporate IR&D in research areas directly feeding NGCR
- Make use of the open systems concept of NGCR to continually transition new computer technology more rapidly into Navy Systems

Approach/Thrusts:

- The Navy strategy for NGCR requires the mix and match of closely integrated systems. This requires considerable research in the area of heterogeneous computing. The application should be relatively unaware of the execution environment. This task attempts to integrate all of the technology areas that contribute to this transparency of computing
- The DARPA Experimental System Kit (ESKIT) will serve as the hardware testbed for integration of busing systems, different processors, networking protocols, operating system interfaces, etc.

Payoff:

- Evolutionary transition of most current computer technology to early Navy use
- Minimal Navy investment in national interest area with maximum results

Major FY89 Accomplishments:

- Established as a Beta test site for DARPA ESKIT
- Participated on NGCR working groups establishing technology needs

Major Users and Related Activities:

- This project is directly targeted at NGCR. Membership on the NGCR committees assures close cooperation and well defined research objectives
- This project will also interface with the JIAWG through NADC
- The output of 6.1 programs in ONR and DARPA are being closely monitored. Primary programs of interest are ONR hard realtime systems in Ada and DARPA ESKIT

C.3.1.3 Navy: Office of Naval Research

Title: Parallel & Distributed Computing Research Programs
Objective: Develop formal scientific underpinnings for design and effective utilization of advanced parallel and distributed computer systems.
Approach/Thrusts:

- Extensive parallel algorithms research
- Software development tools for parallel systems
- Parallel instrumentation techniques

Payoffs:

- Solution to physical limits of uniprocessor computing
- Ability to construct upwardly scalable machines
- Dramatic reduction in run-times for big computations
- Improved dependability of computing facilities

Major FY89 Accomplishments:

- Unity parallel program design paradigm (Misra)
- General techniques for parallel divide-and-conquer (Atallah)
- Parallel algorithms for computational geometry and pattern matching (Kedem)
- Optimal hex-mesh routing algorithm (Shin)
- "SEECUBE" instrumentation software (Krumme)
- "CODE" software development environment (Browne)

C.3.1.4 Navy: Space and Naval Warfare Systems Command

Title: Next Generation Computer Resources (NGCR) Program
Objectives:

- Increase weapons systems operational readiness and interoperability
- Keep pace with rapidly increasing warfare systems processing needs
- Provide the fleet the most effective up to date technology
- Adapt to rapidly changing threat
- Permit competition for product development and system upgrades
- Meet wide range of application requirements
  - Air, surface, sub-surface, land
  - Environment (commercial - full MIL spec)
  - Processing capability
- Have affordable costs
- Increase program manager's flexibility in engineering and applying computer resources

Approach/Thrusts:

- Provides framework for systems design
  - Does not define or standardize on a computer design
  - Standardizes hardware/software interfaces
  - Provides framework for industry Internal Research and Development (IR&D) investment
- Follows standardization trends in commercial sector
- Implemented through joint Navy/Industry working groups
  - Widely used non-proprietary commercial standards base

Payoff:

- Rapid continuous influx of up-to-date technology
- Much larger base of competitive sources
- Leveraging of commercial base
- Interoperability of multi-vendor NGCR compliant products
- Commonality of NGCR products to reduce logistics costs
- Modular/adaptable systems designs
- Program manager flexibility
- Top down weapons system design

C.3.1.5 Air Force: Rome Air Development Center

Title: Parallel Processing Architectures - Application Correlation

Objectives:

- Research the software development process for parallel computer systems and develop a guidebook to direct a software engineer through the parallel software life cycle.
- Provide guidance for matching a particular application/problem to a suitable non-von Neumann computer architecture.

Approach/Thrusts:

- Extend the Phase I study of non-von Neumann computers and software tools; survey parallel computer applications in actual use and determine "real" parallel software design practices and problems encountered.
- Investigate the parallel software development life cycle and analyze the advantages/disadvantages between different software life cycle approaches; analyze each phase of the waterfall model and provide shortfalls, requirements analysis and recommendations.
- Provide a mechanism for determining possible parallel architectures for a given application based upon a number of metrics including physical properties of the hardware, cost, fault-tolerance, software development, processing modes and ease of use.
- Develop a handbook for software engineering and hardware/software selection process; organize information into an interactive database.

Payoff:

- Portray what people in the "real"world do to determine what parallel architecture/mode to chose, based upon availability, preferability, cost, applicability and generality.
- Determine how the waterfall software life cycle can be enhanced or modified to support parallel software development; define steps to consider the various aspects of parallel processing.
- Development and implementation of an interactive database to collectively represent the current state-of-the-art in parallel computing.

Major FY89 Accomplishments:

- Investigations of parallel processing activities commenced with an extensive literature search as well as direct contacts with both developers and users - provided keen insight into many software engineering problems.
- Participated in a workshop on Software Engineering for Parallel Computing.

Major Users and Related Activities: This project is directly targeted at the parallel processing community in general and at researchers/users analyzing parallel architectures.

C.3.1.6 Air Force: Rome Air Development Center

Title: Multiprocessor Software Engineering Methods

Objectives:

- Develop more effective programming methods for asynchronous, parallel multiple-instruction, multiple-data type computers.
- Evaluate these programming methods by developing demonstration programs and compare them to uniprocessor results.

Approach/Thrusts:

- Analyze existing commercial multiprocessors and select the most appropriate multiprocessor-based parallel architecture suitable for medium-grain parallelism.
- Utilize the programming language Joyce, which is a distributed and parallel programming language suitable for medium-grain parallelism.
- Focus on three levels of parallel computing - programming paradigms, parallel algorithms and the degree of parallelism.

Payoff:

- Provide an extensive comparison of different paradigms of parallel decomposition in which processes are organized as either pipelines, trees, rings or meshes.

- Exhibit the effectiveness of supporting Joyce on a parallel machine (Encore Multimax) by writing demonstration programs which implement a multitude of diverse parallel algorithms like sorting, Fast Fourier Transforms, and matrix multiplication.
- Develop programs with different levels of processing size, ranging from one processor to hundreds of processors, and compare results.

Major FY89 Accomplishments:

- Successful implementation of Joyce on the Encore Multimax.
- Selection and utilization of a very powerful medium-grained parallel machine, the Meiko Computing Surface, to exhibit the utilization of Joyce on a different machine.

Major Users and Related Activities:

- Any user who will be developing software for multiprocessor or multicomputer architectures
- This effort will form the basis for a more advanced research effort, "Programming Methods for Multicomputers"

C.3.1.7 Air Force: Rome Air Development Center

Title: Parallel Extensions for Object Oriented Programming

Objectives:

- Examine the current state-of-research in object-oriented programming and develop communication primitives for concurrent languages to support control and data mechanisms.
- Extend analysis of communication calls by means of a formal model called the Synchronous Token-Based Communication State (STOCS) model, an existing tool developed at the University of California at Berkeley.

Approach/Thrusts:

- Provide a survey of the state-of-the-art for both concurrency and synchronization mechanisms in existing programming languages such as Ada, Raddle and Argus and compare with the unit and handshake construct of STOCS.
- Build upon the current STOCS model and remove current limitations that exist for concurrent languages, such as symbolic and induction techniques; investigate other methods for analyzing unit constructs.
- Conducted experiments to show feasibility and productivity gains from using STOCS; evaluate results and provide recommendations to further improve STOCS model.

Payoff:

- Proved that by providing higher level communication primitives in conventional programming languages that the specification of communication aspects can be represented in a realistic manner, thereby extending the real world applicability of concurrent programming techniques in system development.
- Determined how well communication mechanisms and techniques scale up to complex parallel processing domains.
- Extended an existing communication state model to provide parallel extensions for object oriented programming; provided support for hierarchical specification.

Major FY89 Accomplishments: (see Payoff) Effort completed and Final Technical Report submitted.

Major Users and Related Activities: The targeted users of this research are programmers involved in parallel processing wishing to examine communication mechanisms in object oriented programming.

C.3.1.8 Air Force: Rome Air Development Center

Title: S/W Engineering for Parallel Architectures

Objective: To identify the most promising software development tools, techniques and methods that will lead to effective software and system engineering environments for high performance computers.

Approach/Thrusts: A prototype Parallel Evaluation and Experimentation Platform (PEEP) to loosely connect the most promising parallel exploitation tools will be developed. Three Command and Control problems will be implemented using the PEEP and run on a high performance computer. The effort required to develop the software and its resulting quality and performance will be analyzed.

Payoff: PEEP will provide a vehicle for experimenting and evaluating new parallel software technologies. A plan for a cohesive framework of new parallel software development processes capable of producing high quality software products for heterogeneous high performance computers.

Major FY89 Accomplishments: FY90 New Start

Major Users and Related Activities: Any user who will be developing software for parallel, high performance computers. Users who have expressed an interest in the PEEP include RADC/OC and the Avionics Laboratory at Wright Patterson AFB.

C.3.1.9 Air Force: Rome Air Development Center

Title: Parallel Software Performance Predictor

Objectives:

- Develop new software engineering techniques for determining the performance of software for parallel architectures with an emphasis on machines that support BM/C^{3}I applications.
- Design a parallel software perforance predictor software tool to provide explicit estimates of predicted algorithm and software performance parameters for a target parallel machine architecture.

Approach/Thrusts:

- Examine and determine high-level design issues to analyze parallel software requirements; determine and justify selection of target architecture(s)
- Develop a prototype tool(s) that assists in analyzing algorithm designs and associated software, and in predicting their performance for a target computer.

- Design and develop a parallel software performance predictor for a specific target machine within a selected set of architectures; implement prediction tool on Parallel Evaluation and Experimentation Platform (PEEP) at RADC.

Payoff:

- Development of a prototype tool that will assist the systems analyst *in understanding, predicting and effectively utilizing parallel software/ algorithms in parallel processing systems.*
- Prove that a parallel software prediction tool, such as the one to be developed in this effort, can be represented for a specific parallel architecture, thereby extending the real world applicability of parallel processing systems.
- Determine how effective prediction techniques for parallel computing can be accomplished; provide recommendations for future software tools

Major FY89 Accomplishments: None - FY90 new start

Major Users and Related Activities: Software engineers involved in developing software for high performance parallel architectures; users who have expressed interest in a parallel software performance prediction tool include the Avionics Laboratory at Wright-Patterson AFB and the Weapons Laboratory at Kirtland AFB

C.3.1.10 Air Force: Rome Air Development Center

Title: Programming Methods for Multicomputers

Objectives:

- Establish advanced software engineering techniques for providing general-purpose parallel computing; develop a new programming paradigm for establishing and defining new software tools for multicomputers.
- Design and implement a prototype universal parallel programming language for multicomputer processing systems to provide software portability; the key to general-purpose parallel computing is software portability - the ability to execute the same software program on different parallel computers with reasonable efficiency.

Approach/Thrusts:

- Utilize existing parallel programming languages (Occam and Joyce) as the main programming tools to define and implement a prototype universal parallel programming language for multicomputers; devise this prototype language to support software portability.
- Develop a communication harness to use all available processors in a multicomputer system based upon different possible processor configurations that may be employed; devise the harnesses so that during program execution messages between processors are automatically routed.
- Develop and implement a simple and efficient method for load balancing on multicomputers.
- Develop a C^{3}I benchmark application demonstrating the prototype parallel language; investigate the effectiveness and advantages of software portability for parallel processors.

Payoff:

- *Development of a prototype universal parallel programming language* to support the concept of general-purpose parallel computing.
- Prove that the feasibility of designing portable parallel software is realizable, thereby extending the real world applicability of parallel architectures and high performance computers.
- Provide plans for extending software engineering tools to support a new programming paradigm based on this prototype language for multicomputers.

Major FY89 Accomplishments: None - FY90 new start

Major Users and Related Activities: All users involved in developing software for parallel, high performance computers; prototype language will be incorporated into Parallel Evaluation and Experimentation Platform (PEEP) at RADC.

C.3.1.11 Air Force: Rome Air Development Center

Title: Software Techniques for Non-von Neumann Architectures

Objectives:

- Analyze non-von Neumann type architectures for the purpose of determining how both the generic class of these architectures, as well as specific architectures in this class can be utilized over the system and software life cycle
- Determine and assess current software engineering tools and techniques that are applicable to these architectures; determine if new approaches to the software life cycle are required

Approach/Thrusts:

- Survey current state-of-the-art non-von Neumann architectures to identify, characterize and develop a systematic non-von Neumann architecture classification scheme; classify existing machines using this scheme.
- Analyze existing and potential applications for non-von Neumann architectures with an emphasis on BM/C^{3}I, signal processing, image processing, artificial intelligence and real-time simulation; identify problem domains to which non-von Neumann architectures are applicable; provide prognosis of BM/C^{3}I systems in the 1990's.
- Provide software engineering assessment; analyze parallel software tools and parallel software models; determine suitability of software life cycle as specified in DOD-STD-2167A.

Payoff:

- Determined that there currently exists a wide diversity in applications, hardware architectures, computational models, control regimes, programming language constructs and penalties for mismatching for non-von Neumann architectures; *categorized software paradigms and hardware configurations.*
- *Identified that software engineering* for non-von Neumann architectures is severely lacking; there exists limited software tools, very few parallel programming languages and a syndrome that tools will not be built before there are enough machines but not enough machines will be produced until there is enough software support.

- Determined that the current software life cycle is not feasible to support high performance architectures and recommended a prototyping paradigm based on both the spiral model and the STARS model.

Major FY89 Accomplishments:

- Effort completed in Sep 89; Final Technical Report received.

Major Users and Related Activities: The results of this effort are applicable to a multitude of researchers in Government, industry and academia involved in developing software for non-von Neumann computers for applications ranging from graphics to AI.

C.3.2 Communications and Networks

C.3.2.1 Army: AIRMICS

Title: DY10-03/Communications and Networks

Objective: Develop tools, techniques, and prototypes for the design implementation, operation, management, and analysis of systems that are charactarized as heterogeneous networks of networks, incorporating data, voice, video, videotext, and teleconferencing.

Approach/Thrusts:

- Interoperability of Computer Networks
- Integrated Communication Services
- Network Design
- Network Management

Payoff:

- Develop cost-effective, low-risk methods for increasing communication:
  - — capacity — functionality — reliability
  - — responsiveness — maintainability

Major FY89 Accomplishments:

- Electronic Portable Information System Evaluation
- Integrated Services Digital Network (ISDN) R&D Testbed
- Began local area network to ISDN interface development
- Participated in National Science Foundation (NSF) Center
- Completed network assets survey
- Participated in DCA working groups in the areas of Terrestrial Transmissions, Protocol Standards, Defense Switched Networks, Defense Message System, ISDN Standards & Technology, and Integrated Communication Architectures
- Published report on ISDN in the Army

C.3.2.2 Army:Strategic Defense Command

Title: Distributed Intelligent Network Control

Objectives:

- Provide real-time adaptive network control in a changing environment
- Provide the hardware and software necessary for end-to-end network control
- Demonstrate, through experimentation and test, successful end-to-end network control

Approach/Thrusts:

- Develop and evaluate procedures, techniques and algorithms for end-to-end network control
- Construct and provide experiments that employ proven network control technologies to demonstrate and enhance system/subsystem performance

Payoffs:

Support to:

- Network control algorithm/software development
  - — End-to-end control
  - — Routing
  - — Flow-control
  - — Failure recovery and topology update
- Specification development
- Milestone II decision

Major FY89 Accomplishments:

- Defense contract
- Test and development plan

Major Users and Related Activities:

- The system will support the network control experiment
- Other programs which will utilize the results from the program include:
  - — Reliable communications in a perturbed environment
  - — Algorithmic architectures
  - — Test environment support system enhancement
  - — Communications software development

C.3.2.3 Air Force: Rome Air Development Center/DARPA

Title: SDI Network Interface Processing Element/DARPA Research Internet Gateway (SNIPE/RIG)

Objective: Develop a packet switch/gateway for evaluation as a platform for experimentation into high performance networking and for use in operational environments as the next generation packet switch/gateway

Approach/Thrusts: Three identical contracts were awarded to BBN, SRI, and GTE to develop a capability to support SDI and DARPA-related networking experiments as well as operational computer network and internetwork environment. The three designs will be evaluated for their appropriateness for use in SDInet (an SDI testbed) and the Defense Research Internet (the replacement for the ARPANET).

Payoff: Product will be one or more versatile platforms for experimentation by the DoD community into high performance networking as well as the next generation of operational packet switch/gateway.

Major FY89 Accomplishments: Designed, developed, fabricated and tested hardware and software for delivery

Major Users and Related Activities: DARPA is currently replacing the ARPANET with the Defense Research Internet in anticipation of deploying SNIPE/RIGs as the operational gateways. They have also begun to take action to set up an experimental network for DoD-Internet experimentation. These activities will support the Air Force, NASA, and the Dept. of Energy. An SDI research testbed supporting SDI experimentation/validation is being explored.

C.3.2.4 Defense Communications Agency

Title: Packet Switch Verification

Objective: The purpose of this project is to provide a higher level of assurance that DDN software is free of errors that might induce network failures. The DDN provides long-haul data communication between DoD facilities. Unstable or incorrect operation of the DDN could, in certain situations, seriously impact mission essential communications. Recent software induced failures of the Packet Switch Nodes (PSN's) on the MILNET have heightened this need of performing an Independent Verification and Validation (using formal methods) on the PSN software, which will provide a high level of assurance in the reliability of the PSN software.

Payoff: The goal of this project is to use formal methods to provide a higher level of assurance in the PSN software and to introduce formal methods into the PSN software lifecycle process.

Major Users and Related Activities: This project will serve the DDN Program Management Office and be of benefit to all users of the DDN.

C.3.2.5 Defense Communications Agency

Title: Data Communication Protocol

Objective: The purpose of this project is to improve data services of the Internet system. The Internet is the collection of government and research data networks that have evolved from the original ARPANET. This research will identify and develop those government communication capabilities which will not be provided by ISO standards and commercial vendors. Specific objectives are : improve the present quality of data communication services; provide interoperability in the transition period from DoD to ISO protocols; and to ensure that the quality of service required by DoD is maintained with the operation of the new protocol suite.

Payoff: This approach will satisfy the DoD communication requirements at the minimum cost by utilizing commercial products and ISO standards.

Major Users and Related Activities: All of DoD and its related research community.

C.3.2.6 Defense Logistics Agency

Title: Reliable Distributed Transaction Processing

Objective: Investigate protocol issues for reliable transaction processing across distributed heterogeneous systems on an Integrated Services Digital Network (ISDN).

Approach/Thrusts:

- Identify suite of protocols using GOSIP Transaction Processing and ISDN.
- Establish cooperative agreements with the National Institute of Standards and Technology (NIST) and various North American ISDN Users Groups.
- Focus on the following: Transfer of large files (e.g. engineering drawings), low volume transactions (e.g. data base queries), and high volume transaction processing (e.g. Electronic Data Interchange).
- Identify a set of scenarios and conduct a series of investigations to test the viability of each design option.

Payoffs:

- Help prepare DLA for the transition to ISDN.
- Provide a protocol environment for the seamless transfer of data into and our of distributed databases.
- Provide a telecommunications solution for the implementation of CALS and the paper less office.
- Allow for the direct exchange of product data between DLA and the private contracting community.

C.3.2.7 Defense Logistics Agency

Title: Vendor Independent Workstation Protocols

Objective: Identify a standard suite of protocols for connecting a variety of workstations to a variety of data servers and applications.

Approach/Thrusts:

- Define an information transaction protocol that is independent of both the workstations type and the system on which the data resides.
- Define interfaces to be implemented on each type of workstation translating user interaction into a standard information transaction.
- Test the implementation of these protocols in a limited environment of workstation and minicomputer data bases.

- Integrate with the Standard Human/Workstation Interface project.

Payoffs:

- Allows for workstations (both older MS-DOS computers and more capable UNIX systems) to access all applications and systems.
- Makes details about the application (e.g. data location, field names) transparent to the user.
- Allows corporate resources to be managed with minimal disruption to users.

C.3.3 Distributed Operating Systems

C.3.3.1 Navy: Naval Ocean Systems Center

Title: Distributed Operating and Runtime Systems

Objectives:

- Provide a 6.2 focus for the next generation computer resources program (NGCR)
- Stimulate university research, national initiatives and corporate IR&D in research areas directly feeding NGCR
- Make use of the open systems concept of NGCR to continually transition new computer technology more rapidly into Navy Systems

Approach/Thrusts:

- Distributed systems meet the Navy objectives for extension of performance through system upgrades that are relatively painless
- This work consists of influencing existing large efforts. In the area of operating systems the MACH system has been obtained and installed on a Sun/VAX based testbed. The requirements from the NGCR working group on operating systems will be verified and researched using this testbed. Ready systems distributed Ada runtime environment has been obtained and is being installed on this testbed also. This work primarily consists of experimentation to further define requirements and research efforts for industry IR&D investment
- The output of this task will consist of annual reports of progress and attendance at NGCR Working Group meetings

Payoff:

- Evolutionary transition of most current computer technology to early Navy use
- Minimal Navy investment in national interest area with maximum results

Major FY89 Accomplishments:

- Installed MACH on testbed
- Participated on NGCR Working Groups establishing technology needs

Major Users and Related Activities:

- This project is directly targeted at NGCR. Membership on the NGCR committees assures close cooperation and well defined research objectives
- This project will also interface with the JIAWG through NADC
- The output of 6.1 programs in ONR and DARPA are being closely monitored. Primary programs of interest are ONR hard realtime systems in Ada and DARPA ESKIT

C.3.3.2 Air Force: Rome Air Development Center

Title: Distributed Operating Systems

Objective: Develop and demonstrate survivable distributed computing systems composed of multiple clusters of heterogeneous computers interconnected by networks of varying topology and bandwidth.

Approach:

- Cronus enhancements

Payoff:

- Survivability through dispersion and reconfiguration
- Improved crisis management through adaptive resource control
- Interoperability across heterogeneous resources

Major FY89 Accomplishments:

- Tri service distributed system (CRONUS) experiment
- Real time distributed operating system demonstration

Major Users and Related Activities:

- Strategic Defense Initiative

C.3.4 Distributed Database

C.3.4.1 Army: AIRMICS

Title: DY10-04/Very Large Database

Objective: Develop capability to effectively design, implement, operate, and manage very large, truly distributed heterogeneous databases.

Approach/Thrusts:

- Address issues obstructing implementation of distributed heterogeneous database
- Research, design, and prototype an Army Data Encyclopedia
- Implement distributed query processing capabilities

Payoff:

- Identification of encyclopedia specifications

- Identify and address technology barriers prior to operational development
- Reduce risk in moving Army to achieving goal of an Army Information System

Major FY89 Accomplishments:

- Completed initial Army Data Encyclopedia prototype (schema integrator, data dictionary, and browser)
- Natural language interfaces to database systems (report)

C.3.4.2 Air Force: Rome Air Development Center

Title: Distributed Database Integrity

Objective: The objective of this effort is to examine data integrity, consistency and concurrency control techniques for distributed, object-oriented database management systems.

Approach/Thrusts: To investigate the following areas: (1) Rule-based concurrency control techniques for dynamic C^3 processing, (2) Adaptive concurrency techniques for dealing with consistency versus availability of data, (3) Temporal database mechanisms for retaining multiple versions of database objects, (4) Rules, constraints, and application-specific knowledge-bases for integrity maintenance, (5) Active triggers for automatically enforcing integrity rules based upon local or remote updates, and (6) The effect of incrementally mutable objects on distributed database integrity for C^3I.

Payoff: Reliable, distributed database management systems.

Major FY89 Accomplishments: Investigating the advantages and disadvantages of similar concurrency control methods with the goals of method selection and/or transitioning between methods. Currently in the initial stages of developing a framework for multicode consistency intended to directly support tradeoffs between availability and consistency. Investigating which concurrency control techniques can suit object-orientation and developing new techniques as needed.

Major Users and Related Activities: USAF

C.3.4.3 Air Force: Rome Air Development Center

Title: Persistent Data/Knowledge Base

Objective: The objective is to investigate and evaluate persistent object-oriented techniques as mechanisms: (1) for reliable and efficient access to data/knowledge-bases, (2) to increase programmer/user productivity, and (3) to provide powerful modeling techniques in a heterogeneous distributed information processing environment.

Approach/Thrusts: The approach is: (1) To examine advanced integrated heterogeneous networks, (2) Utilize advance storage domains including integrated circuits and optical multi-level storage mechanisms, (3) AI/DB knowledge-centered integration mechanisms, (4) spatial object data/knowledge-bases, and (5) use of natural language interfaces.

Payoff: *Reliable, distributed database management systems.*

Major FY89 Accomplishments: Analyzed the inefficiencies and delays which arise from indirect object accesses. Investigated several approaches for improving system performance. Found the greatest performance improvement would be obtained by delegating control for object accesses to the device drivers, however, further work needs to be done for it to be practical. Investigated ways of improving I/O performance at the storage device based upon storage allocation strategies. Investigated clustering in terms of an object, in order to retrieve an entire object with single disk access, and objects of the same type, to take advantage of typed inter-object references. Investigated methods for specifying indirect object access requirements. Found that pattern-based specification techniques could provide a natural and useful basis for these requirements. Investigated strategies that will support object compaction and garbage collection.

Major Users and Related Activities: USAF

C.3.4.4 Air Force: Rome Air Development Center

Title: Database Techniques for Special Computer Architectures

Objective: Study, investigate, develop and evaluate techniques for high-performance, object-oriented database management systems.

Approach/Thrusts: The approach for this effort began by studying the various types of advanced computer system architectures capable of supporting object-oriented database management system techniques. Architectures to be evaluated include non-Von Neumann architectures, multiprocessors, parallel processors and RISC machines. Also, various memory management, data staging, data placement and secondary storage strategies will be studied. Finally, query processing algorithms will be developed and simulation and modeling of architectural alternatives will be conducted.

Payoff: Techniques, algorithms and guidelines for constructing high-performance, object-oriented database management systems.

Major FY89 Accomplishments: FY90 Start

Major Users and Related Activities: DoD

C.3.4.5 Air Force: Rome Air Development Center

Title: Fault Tolerant Database

Objective: Develop knowledge-based and inferencing techniques to reduce the need for data replication and to enhance data availability and fault tolerance in distributed database management systems (DBMS).

Approach/Thrusts: The approach for this effort began by studying the various types of faults inherent in distributed DBMS environments. A taxonomy of fault types was produced. Next, emphasis was placed upon research applicable to using knowledge-based and inferencing techniques to handle database faults. The purpose of the inferencing techniques is to reduce the extent of data replication (and hence the performance penalties of having to keep replicated data consistent under all conditions), reconstruct lost data or data that are temporarily inaccessible through partition, and resolve inconsistencies during recovery from partition. A demonstration of the technology will occur in Feb 90 for a small application scenario.

Payoff: Techniques and inferencing algorithms for highly-reliable, available distributed DBMS.

Major FY89 Accomplishments: Fault tolerant DBMS design study and logical architecture.

Major Users and Related Activities: DoD

C.3.4.6 Air Force: Rome Air Development Center

Title: Active Data/Knowledge-Base Dictionary

Objective: The objective of this effort is to develop adaptive, robust data/knowledge-base dictionary techniques for integrated C^3I data/knowledge-bases management systems. Areas to be investigated include object-oriented techniques, information resource dictionary standard, persistent object storage and query languages.

Approach/Thrusts: (1) Investigate object-oriented techniques for modeling meta-data and knowledge, (2) Develop techniques for storing characteristics of multi-media data types in the data/knowledge dictionary, (3) Develop techniques for the creation, persistence, and maintenance of objects in the dictionary, and (4) Develop active dictionary components capable of dynamically monitoring and optimizing system performance during processing sessions.

Payoff: Reliable, distributed database management systems.

Major FY89 Accomplishments: This is a relatively new effort, and the contractor has just begun to investigate the impact of using object-oriented organization to represent/describe a data/knowledge-base dictionary system.

Major Users and Related Activities: USAF

C.3.4.7 Air Force: Rome Air Development Center/Strategic Defense Initiative

Title: Real-Time Data Architecture

Objective: Investigate the development of algorithms that provide for time-constrained distributed data management to support SDI BM/C^3 applications.

Approach/Thrusts: Research will be directed towards maximizing both concurrency and resource utilization subject to three constraints at the same time: data consistency, transaction correctness, and transaction deadlines. Issues to be investigated include concurrency control mechanisms, recovery schemes, new data models, parallelism, scheduling algorithms, integration of techniques to handle inconsistencies and condition monitoring, data materialization, garbage collection, data tagging, and update algorithms.

Payoff: The advancement of a unified theory for time-constrained distributed data management.

Major FY89 Accomplishments: Contract award.

Major Users and Related Activities: Immediate customer is SDIO. The end customer is DoD.

C.3.5 Parallel Algorithm Development

C.3.5.1 Navy: Naval Ocean Systems Center

Title: High Performance Computing (HPC) for C^3I

Objectives:

- Provide a vehicle for Navy participation in
  - — High performance computing national initiatives exemplified by "A Research and Development Strategy for High Performance Computing" and "A National Computing Initiative: The Agenda for Leadership"
- Achieve early transition of major research
  - — Breakthroughs in high performance computing to critical Navy applications

Approach/Thrusts:

- Massive parallelism must be used in the future in order to satisfy the requirements for Navy C^3I. This work will include applications development for highly parallel architectures, highly intuitive user/computer interfaces, parallel processing for distributed object oriented databases and C^3I algorithm development
- This work will be targeted to a testbed at CINCPACFLT which is being used to define requirements for the Operation Support System (OSS) system
- Work includes the development of a multiprocessor testbed at NOSC. A number of models have already been converted to the connection machine with resultant speedups of three orders of magnitude over the VAX implementation
- A paper has been developed for presentation at the Supercomputing '89 Conference

Payoff:

- Personnel familiar with new HPC technology will stimulate further advances
- Direct transition to three critical Navy systems (OSS, HIGAIN, FDS)

Major FY89 Accomplishments:

- Converted a number of models to parallel execution of the CM-2 and Encore

Major Users and Related Activities:

- The HPC project is being coordinated with DARPA, NSF, DOE and other national initiatives in high performance computing
- There are three Navy projects currently targeted for transition of HPC products, Operation Support System (OSS), Fixed Distributed System (FDS), and the High Gain Initiative

C.3.5.2 Navy: Naval Ocean Systems Center

Title: High Performance Computing for Undersea Surveillance

Objectives:

- Provide a vehicle for Navy participation in
  - — High performance computing (HPC) national initiatives exemplified by "A Research and Development Strategy for High Performance Computing" and "A National Computing Initiative: The Agenda for Leadership"
- Achieve early transition of major research
  - — Breakthroughs in high performance computing to critical Navy applications

Approach/Thrusts:

- Parallel processing must be used in the future to satisfy the computing requirements of undersea surveillance. The Jason Study on Anti-Submarine Warfare has identified this as a potentially high payoff area for the Navy
- The goal of this task is to develop software packages for spatial processing of large, multidimensional arrays using highly parallel processors. These packages will be utilized for adaptive and non-adaptive processing of data from the tilted array of the HGI Vast-1 experiment, and the array used in the HGI multi-dimensional array experiment. The software packages will provide for dynamic input of arbitrary steering vectors, including near-field focusing and matched field processing with dynamic array element positions.
- The software will be developed initially for the CM-2 and then transported to the ASPEN which is expected to be delivered in December 89. The software packages will be designed for portability which in itself is a research issue considering the radically different HPC architectures.

Payoff:

- Personnel familiar with new HPC technology will stimulate further advances
- Direct transition to three critical Navy systems (OSS, HIGAIN, FDS)

Major FY89 Accomplishments:

- Implemented spatial processing algorithms on CM-2 which will be used to process sea test data from HGI Vast-1 experiment

Major Users and Related Activities:

- The HPC project is being coordinated with DARPA, NSF, DOE and other national initiatives in high performance computing
- There are three Navy projects currently targeted for transition of HPC products, Operation Support System (OSS), Fixed Distributed System (FDS), and the High Gain Initiative

C.3.5.3 Navy: Naval Ocean Systems Center

Title: High Performance Computing for Acoustic FDS Visualization

Objectives:

- Provide a vehicle for Navy participation in
  - — High performance computing national initiatives exemplified by "A Research and Development Strategy for High Performance Computing" and "A National Computing Initiative: The Agenda for Leadership"
- Achieve early transition of major research
  - — Breakthroughs in high performance computing to critical Navy applications

Approach/Thrusts:

- With an increasing fraction of the Soviet submarine fleet expected to be quieter than those we have known in the past, we expect increased reliance on complex surveillance systems that will require interactive graphics to free the operator from data overload
- This task will develop a graphics language as well as build an experimental FDS processing testbed on the connection machine. The primary objective is to develop parallel algorithms and to determine architectural performance measures for real-time multi-beam FDS data processing. The focus is on calculations and development of software algorithms relevant to graphic visualization that allows one to experience the FDS scenarios.

Payoff:

- Personnel familiar with new HPC technology will stimulate further advances
- Direct transition to three critical Navy systems (OSS, HIGAIN, FDS)

Major FY89 Accomplishments:

- Completed Beta release with technical specification for the parallel graphics language (Version 1.0)

Major Users and Related Activities:

- The HPC project is being coordinated with DARPA, NSF, DOE and other national initiatives in high performance computing
- There are three Navy projects currently targeted for transition of HPC products, Operation Support System (OSS), Fixed Distributed System (FDS), and the High Gain Initiative

C.3.5.4 Air Force: Rome Air Development Center

Title: Parallel Problem Decomposition

Objectives:

- Investigate and verify techniques for identifying parallelism as an integral part of the process of problem decomposition.
- Derive a methodological approach to parallel problem decomposition for various types of parallel architectures.

Approach/Thrusts: Little information currently exits on how to design parallel algorithms. In order to be able to effectively utilize commercially available parallel architectures, software must exist which can effectively exploit the computer's potential.

This work will examine, decompose and implement existing algorithms (ex. Radix Sorting) or comprise new algorithms in order to exploit the parallelism intrinsic to the problem. These problems will be exercised on several parallel processing computers.

The output of this task will consist of quarterly progress reports and a final technical report which documents the derived methodology.

Payoff: This work will provide improved techniques for effectively using parallel computer architectures.

Major FY89 Accomplishments: Various algorithms were examined, decomposed and implemented on different parallel processing architectures. A model was developed supporting the derived methodology.

Major Users and Related Activities: This work is targeted to Air Force, private industry and academic organizations who are involved in the theoretical aspects and development of software for parallel processing systems.

C.3.5.5 Air Force: Rome Air Development Center

Title: Strategies for Parallel AI Implementation

Objective: Determine the suitability of existing parallel hardware architectures to support parallel logic programming.

Approach/Thrusts:

- Investigate the suitability of hosting the Knowledge Base Execution System (KBES) on a variety of parallel architectures
- Re-host the KBES simulator to a C implementation to facilitate the transfer to a parallel hardware environment

Payoff:

- A more thorough understanding of the requirements for hosting a logic programming paradigm on a parallel hardware environment.
- Insight into the potential mismatches between language and environment when transferring a non-parallel language onto parallel hardware.

Major FY89 Accomplishments: Investigated and evaluated several architectures as potential hosts for the KBES simulator.

Major Users and Related Activities: Developers of Real-time AI Systems

C.3.5.6 Air Force: Rome Air Development Center/DARPA

Title: Concurrent Expert Systems Architectures

Objectives:

- Discover through experimentation and engineering, all manifestations of parallelism in selected knowledge-based problem solving tasks.
- Develop layered techniques to exploit this parallelism in such a way that the increases in processing throughput combine multiplicatively.

Approach/Thrusts:

- Techniques to support multiple abstraction levels, multiple lines of reasoning and multiple knowledge sources will be developed and implemented.
- Techniques will be tested in a series of "vertical slice" experiments cutting from top user/application level to bottom hardware through several signal processing knowledge-based problem solving tasks.

Payoff:

- Gain a fundamental understanding of the relationship between parallelism and problem solving.
- Design criteria for optimized hardware/software architectures needed in military applications of knowledge-based systems depending for their success on rapid real-time response.

Major FY89 Accomplishments:

- Software systems for measuring multiprocessor applications performance for a range of system architectures using alternative programming paradigms.
- Concurrent objective programming methodology and interface evolved through real applications to represent useful and efficient parallel programming model.

Major Users and Related Activities:

- Real time AI system designers
- Knowledge base system developers
- Under DARPA's Strategic Computing Program

C.3.5.7 Strategic Defense Initiative/Army: Strategic Defense Command

Title: Algorithmic Architectures

Objectives:

- Optimize BM algorithms and processor architectures
- Maximize BM high performance data processing
- Provide advanced experimental approaches with proven algorithm technologies

Approach/Thrusts:

- Acquire BM algorithms
- Develop BM algorithms as required
- Evaluate BM algorithms and emerging computer hardware

- Optimize the algorithms to the hardware
- Demonstrate/evaluate advanced BM data processing performance

Payoffs:

- Support to:
  - — BM/C^3 testability
  - — Fault tolerance
  - — Birth-to-death tracking
  - — Battle group formation
  - — Dynamic strategy implementation
  - — Experimental version performance assessment (EVPA)
  - — Algorithm complexity
  - — Security
  - — Threat tube formation
  - — Discrimination data fusion
  - — BM technology

Major FY89 Accomplishments:

- Award of contract
- Define system requirements

Major Users and Related Activities:

- Algorithmic architectures program is the evolutionary follow-on to the multiple information set tracker correlator and the weapon target assignment programs, building on the foundation laid by these and other efforts
  - — SDS System Engineer Contractor
  - — NTF/ARC/NTB
  - — EVPA
  - — Other government agencies and contractors

C.3.6 Parallel Processing Languages

C.3.6.1 Strategic Defense Initiative

Title: Parallel Computing for SDI Battle Management

Objectives:

- An easy-to-use, portable way to program parallel computers and to focus the power of an entire network on a single massive problem.
- Producing efficient code for large-scale distributed memory machines
- To obtain high performance from the inner loops and computational kernels of programs.

Approach/Thrusts:

LINDA:

- Portable system for programming parallel machines and coordinating networks
- Converts heterogeneous networks into a single, integrated powerful computing environment
- Adapts to any host language: Ada, Fortran, C, etc.
- Provides a simple, powerful, language independent methodology for programming parallel computers

CRYSTAL:

- Very high-level language and compiler for distributed memory machines
- Contains techniques and tools for manipulation and global optimization of parallel computation
- Optimizations are based on the notion of reshaping of index domains, addressing the data movement and distribution problem for large scale multiprocessors
- Metalanguage environment supports interactive parallel program transformation and user-specified algorithmic problem partitions

PARTY:

- Seeks methods for run-time performance optimizations
- Uses information available during program execution to rearrange the order work is performed to increase exploitable parallelism
- In machines with strong memory hierarchies, additional optimizations include:
  - — Effective problem partitioning
  - — Effective use of local memories to cache information
  - — Reduction of communication startups by clustering data to be communicated

Payoffs:

- LINDA simplifies parallel program generation and subsequent control of homogeneous (all the same type of computer) and heterogeneous (different types of computer) parallel processing systems with complex realtime problems. This can be done in the context of conventional programming environments.
- Compiler technology for distributed memory machines and shared memory machines with memory hierarchy. Automatic techniques for parallel program generation and optimization beyond the Fortran parallelizing preprocessor. Fast prototyping for design of special purpose processors such as those in signal processing applications. Interactive parallel program transformation and support for user-specified algorithmic problem partitions.
- Substantial improvement in the ability of parallelizing compilers to optimize different types of loop structures. General purpose mechanisms for optimizing the performance in important classes of computational kernels on distributed memory machines and in machines with strong memory hierarchies. Facilities for programmers to easily specify problem-dependent partitioning strategies and a standard set of runtime problem mapping modules available to users.

Major FY89 Accomplishments:

- Generated a customized support library and new compile-time optimizer in C for C LINDA on Encore and Sequent Systems
- Alpha version of compiler for Intel IPSC/2 & NCUBE
- Demonstration Fortran preprocessor for a limit class of "DO CONSIDER" loops for ENCORE parallel Fortran.

C.4 Real-Time/Fault Tolerant Computing

C.4.1 Army: CECOM, Center for Software Engineering

Title: Ada Real-Time/Runtime (A094 Tactical Software Engineering Technology)
Objective: Explore Ada real-time/runtime technology and provide guidance for the development of embedded real-time Ada systems.
Approach/Thrusts:

- Develop concept for near-term workarounds to critical real-time problems and research long-term solutions
- Base program on recognized problems and consensus of experts
- Use proof-of-concept experiments to test proposed methods, guidelines, and concepts
- Coordinate and be aware of current work by other organizations
- Hold regular technical interchange meetings

Payoff:

- Guidelines to select, configure, and use an Ada runtime environment
- Guidelines to tailor a runtime environment
- Benchmark evaluation, specification, and mapping to real-time requirements
- Study of problems, related methods, and solutions
- Guidelines for transportable real-time Ada software
- Distributed Ada real-time software to improve performance using the Ada tasking model with rendezvous as the communication mechanism
- Technology transfer through published papers, conference presentations, and technical interchange meetings

Major FY89 Accomplishments:

- Real-time Ada problem solution study
- Guideline to select, configure, and use an Ada runtime environment
- Catalogue of Ada runtime implementation dependencies
- Real-time Ada demonstration project
- Transportability guideline for Ada real-time software
- Methodology framework for the development of real-time Ada software
- Analysis of the impact of the Ada runtime environment on software reuse
- Software first system development methodology
- Approach to tailoring an Ada runtime environment
- Analysis of Ada tasking performance issues for uniprocessor and distributed applications
- Analysis of scheduler issues for real-time applications
- Development of process to produce composite benchmarks that measure performance
- Analysis of runtime interfaces and Ada/Posix bindings with real-time extensions
- Method for specification and verification of timing constraints for Ada embedded real-time systems

Major Users and Related Activities:

- Direct consulting to Center for Software Engineering who support various systems
- Direct and indirect consulting to army PMs and PEOs
- Exchange of information with other organizations, e.g. Airmics, Software Engineering Institute, NATO.

C.4.2 Navy:Office of Naval Research

Title: Dependable & Real-Time Computing Research
Objective: Develop scientific underpinnings for design and construction of dependable time-driven computing systems.
Approach/Thrusts:

- Formal specification and verification techniques
- Hard real-time resource management theory
- Fault trend prediction techniques
- Dependable multi-computer system techniques

Payoffs:

- Ability to construct real-time systems with predictable behavior
- Substantial reduction in system testing phases
- Greatly simplified system enhancement
- Certifiably dependable automated systems with non-trivial functionality

Major FY89 Accomplishments:

- Initiation of 5-year research effort on foundations of real-time computing
- Dramatic expansion and improvement in academic research on real-time computing
- Transition of reliable systems workbench to DoD and industry

C.4.3 Air Force: Wright Research and Development Center/Avionics Laboratory

Title: Avionics Fault Tolerant Software
Objective: Develop and demonstrate software fault tolerant techniques for real-time mission-critical software, not

including N-version programming or recover block techniques.

Approach/Thrusts:

- Examine requirements for real-time avionics fault tolerance.
- Develop and demonstrate techniques to support fault tolerance including a software test and maintenance bus.
- Examine using the Built-In-Support function developed under Embedded Computer Support Improvement Program (ESIP).

Payoffs:

- Software fault tolerance for real-time, mission critical software will be addressed and applied.
- Burden of support organizations in solving intermittent timing and data errors will be reduced.
- Fault tolerant software.
- Increased mission-completion-success probabilities for weapon systems.
- System reliability will improve.

Major FY89 Accomplishments:

- New start in FY89.

C.4.4 Air Force: Rome Air Development Center

Title: Survivable Adaptive Planning Experiment

Objective: Demonstrate survivability to post-SIOP and build.

Major FY89 Accomplishments:

- Phase I Three Contractor Design Study Completed.
- Phase II Contract Awarded.

Major Users and Related Activities:

- Major users
  - — Joint Strategic Planning Staff (JSTPS)
  - — Strategic Air Command (SAC)
  - — Naval Ocean Systems Center (NOSC)
  - — Army Communications Electronics Command (CECOM)
  - — Strategic Defense Initiative
- Related activities
  - — DARPA
  - — ARMY/CECOM
  - — AF/WRDC
  - — SDIO
  - — NSA
  - — NAVY/NOSC
  - — NASA

C.4.5 Air Force: Rome Air Development Center

Title: Fault Tolerant Design

Objective: Develop a formal software fault tolerant (SWFT) development methodology for the generation of designs responsive to stringent reliability requirements.

Approach/Thrusts: Involves completion of two tasks:

- Task I: Develop and analyze the requirements for the SWFT development methodology components, and produce a top level design for each. These components will be (1) a SWFT System Description Model (SDM) Man-Machine Interface (MMI) which will be used to generate SWFT designs for a particular application system, (2) a Reliability Prediction Module (RPM) which will be used to predict the reliability of those SDM designs so as to meet the requirements of that application system, and (3) a Cost Benefit Analyzer to determine the most cost effective design of those generated to use for the application system.
- Task II: A prototype SDM MMI and RPM will be developed to demonstrate feasibility and proof-of-concept.

Payoff: A prototype software system to generate designs from requirements using techniques appropriate for SWFT application systems.

Major FY89 Accomplishments: The requirements analysis for a Computer-Aided Software Engineering (CASE) tool, called the Software Fault Tolerant Design System, which consists of the MMI, RPM, and Cost Benefit Analyzer components, was completed, and the top-level designs for each component were documented in a final technical report.

A prototype tool was developed from the top-level design, in Ada, on an IBM-AT. The prototype consists of a MMI and a RPM, and uses a Markov Model approach to SDM definition, and Kronecker algebra techniques for reliability predictions of designs based on SDM's.

Major Users and Related Activities: Any software development where software fault tolerance is a required system characteristic.

C.4.6 Air Force: Rome Air Development Center

Title: Software Engineering for Fault Tolerant Systems

Objective: Evaluate current state-of-the-art in software fault tolerance; identify gaps in the technology; identify software engineering technology needed to support software fault tolerant systems development and to provide recommendations for research and development in fault tolerance and software engineering technology.

Approach/Thrusts: The approach consists of an extensive literature search and personal contacts with researchers working on fault tolerant approaches for sequential software, real-time applications and distributed systems. Software engineering technology needed to support the development and application of fault tolerant techniques in software systems will be investigated.

Major FY89 Accomplishment: Technical effort was completed. Final technical report will be delivered in early FY90.
Payoffs: Current assessment of the state-of-the-art and state-of practice of software fault tolerance and supporting software engineering technology. Recommendations for specific RD activities to advance the application of fault tolerant techniques in software systems and to provide software engineering tools and techniques that support fault tolerant applications.

C.5 Secure/Trusted Software and Systems

C.5.0.1 Navy: NCSC/SPAWAR

Title: Computer Security (COMPUSEC) R&D
Approach/Thrusts:

- Trusted Data Base Management Systems
- Higher Order Language Analysis
- Secure Software Engineering
- Risk Modeling and Analysis
- Hardware Verification

C.5.0.2 DCA: Center for Command, Control, and Communication Systems

Title: Joint Multilevel Security (MLS) Technology Insertion Program
Objectives:

- Identify emerging MLS technologies and insert in command and control systems development.
- Field limited MLS capability by end of FY 1991.
- Expand capability in out years.

Approach/Thrusts:

- DCA is the Program Manager.
- The Joint Staff sets policy, coordinates and prioritizes requirements.
- The National Security Agency (NSA) provides technical guidance.
- Two CINC initiatives have been approved for prototyping using emerging contractor-developed MLS technologies; additional command prototype efforts are anticipated.

Payoffs:

- Rapid prototyping and evolutionary insertion of emerging MLS technologies into command and control systems will make MLS capabilities available to commands sooner.
- This approach will minimize costs by leveraging command initiated efforts and by sharing Service developments.

Major Users and Related Activities:

- Prototype initiatives have been approved at the Military Airlift Command and the US Central Command.
- Primary beneficiaries will be the unified and specified commands and major component commands.

C.5.0.3 Defense Logistics Agency

Title: Security and Data Protection in CALS
Objectives: Investigate methods to determine data aggregation risks using AI examination of CALS queries.
Approach/Thrusts:

- Develop a series of algorithms to determine the security risks associated with data aggregation.
- Develop the capability to determine security risks directly from query entries.
- Implement and test prototype system.
- Evaluate results of prototype system and project the costs and benefits of implementing the system in a production environment.

Payoffs:

- Identify methodology for assessing security risks from aggregation of data.
- Propose methods for identifying and handling risks as they occur operationally.
- Increase the acceptance of the CALS initiative by providing DoD with the capability of identifying security risks and upgrading protection of DoD and vendor proprietary data.

C.5.1 Secure/Trusted Development Environments

C.5.1.1 Strategic Defense Initiative/National Security Agency

Title: Trusted Software Environments
Objective: To produce a trusted software engineering environment (SEE) for the SDS Software Center. The Software Center environment will "showcase" security features and functionality to SDS elements.
Approach/Thrusts: The System Engineer will upgrade the initial SC SEE to meet the security requirements. The approach is to develop a prototype SEE and assess its capabilities.
Payoff: The development of a trusted SEE to be used in the development of the trusted software required for the SDS.

C.5.2 Development Techniques for Secure/Trusted Software

C.5.2.1 Air Force: Rome Air Development Center/Strategic Defense Initiative

Title: Assured Service Concepts and Models
Objective: Advance the theory of assured service for distributed systems.
Approach/Thrusts: This research shall include:

- Identification of issues and approaches regarding detection, recovery, and prevention of denials of critical services in a multilevel secure distributed system.
- Development of models to specify availability, reliability, survivability, and performance of critical services.
- Formalization of the concepts regarding adaptive security policies for multilevel secure distributed systems, including the development of models to permit reclassification of information, operational mode changes, reconfiguration of system resources, concatenation of differing component system policies, and broadcast messages.

Payoff: The formalization of concepts for assuring service in distributed systems to permit inclusion of the requirement in future systems design.

Major FY89 Accomplishments: FY90 new start

Major Users and Related Activities: Immediate customer is SDIO. The end customer is DoD.

C.5.2.2 Air Force: Rome Air Development Center

Title: Malicious Code in Distributed Systems Analysis

Objective: Investigate vulnerabilities in distributed systems' design to malicious code and identify policy, mechanism, and assurance countermeasures to ensure confidentiality, integrity, and service.

Approach/Thrusts: This investigation shall include:

- Identification of the architectural and functional weaknesses in distributed systems, such Cronus, Alpha, and MACH, that make them susceptible to malicious code, including viruses, worms, trojan horses, logic bombs, and time bombs.
- Identification of security policy, hardware and software mechanism, and assurance countermeasures for each of the weaknesses/vulnerabilities identified above to ensure confidentiality, integrity, and service.
- Analysis of the extent to which the experimental Secure Distributed Operating System protects against the vulnerabilities identified.
- Development of a secure distributed system designer's handbook of distributed system architectural features and functions and corresponding security policy requirements, security mechanisms, and assurance features for ensuring confidentiality, integrity, and service, even in the presence of malicious code attacks, for various modes of operation and environments.

Payoff: Evolutionary development of more highly trusted distributed systems; minimum investment with maximum results.

Major FY89 Accomplishments: FY90 new start

Major Users and Related Activities: Electronic Security Command is the immediate customer. The end customer is DoD.

C.5.2.3 National Security Agency

Title: Secure Software Engineering Technologies

Objectives: Secure software engineering is now disjoint from the system acquisition and life cycle process. Security engineering tasks, as defined in DoD5200.28-STD are not integrated into the system acquisition life cycle, as exemplified in DoD-STD-2167A. Security concepts related to correct system behavior are not formalized in Ada-based architecture design languages. This results in the inability to adequately express required security concepts and the inability to apply design simulation technologies to the more advanced, secure systems. Additionally, there is very little interaction in the computer industry toward proper and logical development of efficient and effective software engineering, much less secure software engineering. This program will lead the industry toward a coordinated and consolidated effort that will benefit all concerned; including the Agency. Unless we can capitalize on the present window when vendors are struggling to create more complex software systems, we will miss out. The opportunity cost right now is infinitesimal when compared to leveraging a burgeoning industry in the future.

The goal of this program is the development of methods, techniques, and tools to provide for the development and maintenance of secure software systems. To that end, this program is focused on: the development of a mathematically based definition of trust and trust metrics, the infusion of formal methods into software engineering and the development of environments that provide for rigorous control of software systems developed on distributed systems. In direct support to the DoD, this program also addresses the concerns of using the Ada language for secure system development. The overall objective is to develop guidelines, models, and worked examples which will facilitate the integration of security requirements into the system life cycle (for systems such as the Strategic Defense System) in such a way that an independent security accreditor/ evaluator can validate that the security requirements are indeed implemented in the system.

Approach/Thrusts: The program contains six (6) projects (Formal Methods Infusion, Ada for Secure Software Engineering, Software Configuration Control and Management,Trust Foundations, Software Engineering for Secure Applications, and Executable Specifications), each with multiple tasks.

- Formal Methods Infusion: Our goal is to foster the use of formal methods throughout the life cycle to significantly increase the assurance in systems. Without the use of formal methods, the best software engineering efforts can only provide an intuitive, hueristic belief in the security of a system. Our approach is three-fold: 1) to extend current, production quality tools to incorporate formal methods; 2)to work with the Software Engineering Institute (SEI) and universities to develop curricula modules and support tools; and 3) to facilitate the integration of formal methods into software engineering.
- Ada for Secure Software Engineering (formally Ada Assurance): The Ada language is the language of choice for the DoD (DoD Directive 3405.1) and the required language for weapon systems (DoD Directive 3405.2). To further the

assurance provided to the Ada-based secure systems, many areas of the Ada language need to be investigated. This project area deals with the issues of a trusted Ada compiler and a trusted Ada runtime environment with mathematically precise semantics,programming guidelines and security preprocessors.

- Software Configuration Control and Management: The development of large software systems is performed on a heterogeneous, distributed hardware environment. Controls on a system being developed are ad hoc, at best, and do not provide the rigor necessary for secure mission critical systems. This research project is investigating the development of life cycle assurance for secure software systems. Initial work focuses on centralized controls for secure systems development, and then will distribute these controls to provide a secure, integrated, distributed environment for software systems development. This work includes the development of a life cycle control framework, a distributed object model and management system, a heterogeneous hardware testbed, and a trusted distribution capability.
- Trust Foundations: This research project is focused on developing a formal definition of trust, software metrics based on the definition, and tools that generate these metrics. Initial work is being done under the SDI-Focused Trusted Software Project (TSP). Whereas the TSP is focused on the short term initial efforts; this project will build on lessons learned to provide a sufficient base for establishing the validity of the definition and metrics. This will provide a more generic solution for the DoD community.
- Executable Specifications: Formal specification techniques can provide a rigorous foundation for building secure computer systems. However, the formal specification languages that are the most amenable to rigorous proof methods are difficult to write and interpret. Proving that a specification is consistent with a formal mathematical security model may assure its security properties but cannot assure that the specification captures the user's intentions for system functions.

  One solution to this problem is to develop a means of writing executable specifications so that both the system designer and the implementor can test the specification to what is intended by the specification.

  Specifying software systems with traces has benefits both from a software engineering perspective and from a verification perspective. Traces are used by Hoare in his development of Cooperating Sequential Processes, and a formal semantics for trace specifications has been published by McLean. Preliminary work at NRL has demonstrated the feasibility of executing trace specifications for small example systems directly against his expectations prior to undertaking rigorous formal proofs. The current translator also permits testing a specification for inconsistencies. The goal of this project is to extend the existing tools for executing relatively small example specifications to permit the execution of large scale trace specifications.

  The Navy's approach will be to use this prototype, a relatively small-scale translator, as a basis for building a translator that will accept specifications of sizable practical systems.
- *Software Engineering Techniques For Secure Application* System Development: It is commonly agreed that using better software engineering methods, with early attention to system security requirements, leads to more secure computer systems. The TCSEC provides some guidance on appropriate design methods to be used in developing Trusted Computing Bases. Lacking, however, is a well-defined, well-integrated, and well-documented method for incorporating software engineering and security requirements in the development of application systems. Further, software written for the Department of Defense must adhere to strict software development standards that affect the entire software life cycle, from design to maintenance. Unfortunately, many of these standards do not recognize the special needs of computer security. In some cases, they may even hinder the development of secure systems by imposing outdated software engineering technologies.

  The effort under this project will be to examine existing DoD software standards and current software engineering methodologies to assess their ability to support development of secure applications systems.

  Incorporation of the best software engineering practices in the development of application systems has long been a focus of NRL's Secure Military Message Systems (SMMS) project. The SMMS project will yield a worked example of how to develop and document application system software with significant security requirements. In addition, NRI has been directly involved in the evaluation of application systems produced by major defense contractors. Efforts under this proposal will incorporate this experience and yield reports concerning:

  — How existing software standards can best be used to support secure system development and where those standards need to be modified.
  — An overall approach to software engineering for secure application systems.
  — Areas in which formal techniques can be applied effectively.
  — Areas in which automated support would be feasible and useful.

Depending on the outcome of these studies, follow-on work to identify or develop particularly appropriate software development tools or environments may be proposed. The proposed budget anticipates development of some such tools.

C.5.2.4 National Security Agency/Strategic Defense Initiative

Title: SDI-Focused Research

Objective: This SDI-Focused research is comprised of several significant elements and includes the Trusted Software Project, Design Guidance and the SDI Operations Battle Management Processor. This program is separately funded by SDS, and complements the program mentioned above. Currently, there is a very limited manufacturing base to support the secure development of an SDS. Vendors are not producing the products or working on this technology for the future. Therefore, with very limited federal funding, progress has been extremely slow. Ultimately, what the National Computer

Security Center desires is for the SDS to be developed (through its two decade life cycle) on secure machines using trusted software. We are working toward this end.

Approach/Thrusts: Research areas include: trust foundations (definitions, metrics and tools), trusted distribution strategies, software methodology and library vulnerabilities. V45 has managed (for SDIO), a contract for initial work in this area by a team led by GE.

The program contains three (3) projects (Trusted Software Project (TSP), SDI Operations Battle Management Processor, and Secure Engineering Integration R&D.

C.5.3 Secure/Trusted Operating Systems/Compilers

C.5.3.1 Army: CECOM

Title: Army Secure Operating System (ASOS)

Objective: To develop a family of secure, real-time operating systems. The ASOS program has resulted in a dedicated secure operating system certifiable to the C2 level, and optimized for efficiency, and a multilevel secure operating system certifiable to the A1 level, optimized for security.

Approach/Thrust:

- Design to efficiently support real-time tactical systems.
- Provide a scheduler for both the C2 and the A1 system implements the Ada scheduling rules and use target hardware to efficiently switch tasks.
- Provide a priority preemptive scheduler as well as a deadline scheduler.
- Program and task priorities support.

Payoff:

- ASOS provides a flexible system generation facility through which a desired combination of security and real-time features can be selected.
- The multilevel and dedicated secure versions of ASOS are designed specifically to efficiently support Ada Applications.

Major FY89 Accomplishments:

- An A1-certifiable ASOS system, running on a Sun-3 computer, has been delivered to the Army.
- Copies of ASOS are being made available to the Naval Research Laboratory and Rome Air Development Center to encourage interservice cooperation, usage, and experience with ASOS.

C.5.3.2 Air Force: Rome Air Development Center/NSA

Title: Experimental Secure Distributed Operating System Development

Objective: Investigate distributed system multilevel security issues as they relate to distributed operating system design and implementation.

Approach/Thrusts: Using the Cronus Distributed Operating System as a baseline, the contractor will implement and demonstrate a trusted distributed operating system capability that meets B2 assurance criteria as a minimum, with B3 as the goal. The output of this effort will be a significant subset of 2167A documentation and a feasibility demonstration.

Payoff: Evolutionary development of trusted distributed systems; minimum investment with maximum results.

Major FY89 Accomplishments: Produced software development plan, system software specification, software requirements specification, system/segment design document, and formal security policy model. Participated in (Navy) Next Generation Computing Resources Operating Systems Standards Working Group.

Major Users and Related Activities: The National Computer Security Center (NSA/C3) is the immediate customer. The end customer is DoD. Participated in (Navy) Next Generation Computing Resources Operating Systems Standards Working Group.

C.5.3.3 Air Force: Rome Air Development Center/NSA

Title: Distributed Trusted MACH Concept Exploration

Objective: Investigate trust issues of extending the trusted MACH (TMach) operating system to operate in a distributed fashion over a heterogeneous collection of machines.

Approach/Thrusts: This task shall include the examination of the TMach Name Server/Net Server interaction locally and across the network; issues of object name space, user and machine identification/authentication, exploitable covert channels, and configuration control; modifications required to other trusted TMach servers for them to function in a distributed system; issues associated with strategies for data replication within the distributed system; audit across the distributed system; and the use of encryption as a technique to facilitate system authentication as well as to protect transmissions.

Payoff: Evolutionary development of trusted distributed systems; minimum investment with maximum results.

Major FY89 Accomplishments: FY90 new start

Major Users and Related Activities: The National Computer Security Center (NSA/C3) is the immediate customer. The end customer is DoD.

C.5.3.4 Air Force: Rome Air Development Center/DARPA

Title: Information Security (INFOSEC) Workstation

Objective: Develop an information security (INFOSEC) workstation that integrally combines trusted operating systems technology with communications security technology and provides sophisticated access and integrity controls required for mission critical interoperability.

Approach/Thrusts: Modify the trusted MACH operating system (a B3 level system) to incorporate NSA-approved cryptography, such as Tepache, into the communications protocol structure. Access and integrity controls will also be enhanced. As an option, the contractor will formally verify the correct operation of the DDN Packet Switch Node code that provides the basic network communications services.
Payoff: Multilevel Secure (MLS) workstations with embedded cryptography for MLS intercommunications, enhanced access and integrity controls; evolutionary development of more highly trusted distributed systems; minimum investment with maximum results.
Major FY89 Accomplishments: FY90 new start
Major Users and Related Activities: DARPA is the immediate customer. The end customer is DoD.

C.5.3.5 National Security Agency/Air Force: Rome Air Development Center

Title: Distributed System Security
Approach/Thrusts:

- Secure distributed operating systems
- Secure database management
- Tools and methodologies

C.5.3.6 Strategic Defense Initiative/National Security Agency

Title: Trusted Ada Compiler and Runtime Environments
Objective: To develop a trusted Ada compiler and runtime environment which will be transparent to the secure operating system's hardware base and will meet the SDS' security and performance needs.
Approach/Thrusts: To perform preliminary studies which will assess the inherent vulnerabilities of these tools, survey existing technology and outline SDS requirements. The trusted tools will then be produced by companies with previous experience developing quality Ada compilers and runtime environments.
Payoff: To develop the ability, through the use of trusted tools, to produce the trusted Ada software that will be required for the SDS.

C.5.4 Secure/Trusted Databases

C.5.4.1 Air Force: Rome Air Development Center

Title: Secure DBMS Auditor
Objective: To develop techniques which enable the collection and processing of reliable audit data in a trusted database management system (TDBMS) application environment.
Approach/Thrusts: The approach for this effort began by studying the various types of auditable events in TDBMS application environments. The auditable events were then studied in relation to various and differing TDBMS architectural approaches. Site surveys were conducted in order to gather realistic information on the methods used in TDBMS audit systems. Finally, audit system deficiencies and a functional specification of a TDBMS audit system were produced.
Payoff: An increased understanding of both the auditing process in TDBMS application environments and the relationship of auditing to the process of trusted system evaluation in accordance with the DoD Trusted Computer System Evaluation Criteria.
Major FY89 Accomplishments: Functional specification of a TDBMS auditing system and architecture.
Major Users and Related Activities: DoD

C.5.4.2 Air Force: Rome Air Development Center/NSA

Title: Secure Knowledge Base System
Objective: The objective of this effort is to develop a security policy and formal model for a knowledge-base system.
Approach/Thrusts: The approach for this effort is to study and investigate the multilevel-security (MLS) requirements for knowledge-base systems. After the requirements analysis is completed a security policy and formal, mathematical security model will be developed for the system.
Payoff: Initial study of security requirements for knowledge-base systems.
Major FY89 Accomplishments: Object-oriented model chosen as the data model for the secure knowledge-base system.
Major Users and Related Activities: DoD

C.5.4.3 Air Force: Rome Air Development Center/NSA

Title: Trusted DBMS Prototype
Objective: Develop a DoD Trusted Computer System Evaluation Criteria A1-compliant database management system (DBMS) prototype based upon the Secure Distributed Data Views (Seaview) system design.
Approach/Thrusts: The approach for this effort is to construct a detailed design of the system based upon the Seaview security policy, formal model, formal top level specification and implementation-level specification. The TDBMS prototype will utilize commercial-off-the-shelf components such as the Oracle DBMS and Gemini A1 computer. Additional software will be coded, tested and integrated in accordance with DoD software standards. The prototype will then be evaluated based upon security, performance, reliability and user-friendliness factors. Delivery of the prototype is expected around Dec 91.
Payoff: A prototype A1 compliant trusted, relational DBMS.
Major FY89 Accomplishments: Software Development Plan and System/Segment Specification.
Major Users and Related Activities: DoD

C.5.5 Secure/Trusted Communications Networks

C.5.5.1 Strategic Defense Initiative

Title: SDI Network Security

C.5.6 Standards for Secure/Trusted Systems

C.5.6.1 Navy: Space and Naval Warfare Systems Command

Title: ADP Security

Objectives:

- Develop products to enhance security in future operational systems
- Develop security analysis methods and models
- Develop security guidelines and procedures for Open System Architecture Interface
- Develop and verify standards for integration of Trusted Information Components

Approach/Thrusts:

- Develop trusted components
- Develop standards for trusted Open system Architecture
- Develop and verify integration standards and guidelines
- Develop certification guidelines
- Showcase trusted system models

Payoff:

- Detailed security guidance for developing Multi-Level Security in Open System Architecture
- Methods and models for use in implementing certified trusted application systems
- Vehicle for on-going evaluation and integration of security products and techniques for Navy C^3I systems

Major FY89 Accomplishments:

- Prototype Intrusion Detection Expert System for monitoring Sun system activity

Major Users and Related Activities:

- Users
  - — Navy program/project managers with computer security requirements
  - — Contractors developing Navy systems
- Related activities
  - — NSA (NCSC)
  - — JLC
  - — DCA
  - — Army
  - — Johnson Space Center Study Group
  - — Air Force

C.5.7 Certification/Verification Techniques for Secure/Trusted Systems

C.5.7.1 Navy: Office of Naval Research

Title: Formal Foundations of Secure Systems

C.5.7.2 National Security Agency

Title: Significant Verification Examples

Objective: As computer technology continues to proliferate and the rate of innovation in applying that technology accelerates, Information Security (INFOSEC) depends more and more on the highest levels of trust, as outlined in the Trusted Computer System Evaluation Criteria (TCSEC or "Orange Book"). The current highest level of trust, A1, depends upon the successful application of formal methods of analysis and system derivation through the use of an approved verification environment, or verification tool. Both the state-of-the-art in verification tools and useful results from the application of those tools suffer from a "catch 22" situation. Potential users of the existing formal tools are either reluctant or incapable of successfully applying them to realistic problems. Tool developers are at a loss for meaningful beta tests of their products and feedback on enhancements which will make those tools "production quality".

Approach/Thrusts: This program provides computer system design engineers with the appropriate software verification tools to develop computer systems for DoD at the A1 level of assuredness. Lessons learned will feed "advanced formal methods research" developments and guide our strategy to sell formal verification to the world of software engineering. The program contains five (5) projects (Current Endorsed Tools List Examples, Existing Tool Examples, Advanced Tool Examples, Communication and Data Storage System, and VALKYERIES) some with multiple tasks.

C.5.7.3 National Security Agency

Title: Verification System Enhancements

Objective: DoD has a need for computer systems that meet A1 level of trust. System formal top level specification verification is required to satisfy the A1 criteria. This program provides the necessary research and development to equip the computer system development community with a verification environment to appropriately design A1 systems. Through research we will produce enhanced verification environments, using the results of advanced formal verification methods, and investigate a number of promising verification techniques.

This program provides computer system design engineers with the appropriate software verification tools to develop computer systems for DoD at the A1 level of assuredness. The program contains four (4) projects (Gypsy Verification Environment, Formal Development Methodology, Enhanced Hierarchical Development Methodology and Formal Top Level Specifications (FTLS)-Based Testing) some with multiple tasks.

C.5.7.4 National Security Agency

Title: Advanced Formal Methods Research
Objective: The objective of this program is to advance the state-of-the-art in computer security verification and modeling techniques and environments to provide assured systems are obtained within the framework of advanced computer hardware architectures and new software engineering techniques. This functional area will: conduct technical interchanges; provide technical guidance; coordinate activities among CCSP participants, contractors, and academia to assist developers in producing verification environments using advanced formal methods; provide basic research in advanced formal verification methods; and foster the injection of advanced verification technologies and techniques into the emerging discipline of software engineering.

This program will provide computer system design engineers with the appropriate software verification tools to develop computer systems for DoD using advanced formal methods for establishing increased level of assuredness. It includes advancements in mathematical modeling, research into extending formal methods deeper into software and system paradigms, researching issues related to data integrity and denial of service, and investigations into the validity of the current computer system security engineering paradigm.
Approach/Thrusts: The program contains three (3) projects (Defining Advanced Formal Methods Research Environments, Advanced Verification Environment Research, and Advanced Security Modeling Research) each with multiple tasks.

C.5.7.5 National Security Agency

Title: Integrated Assurance Techniques
Objectives: The objectives of the Integrated Assurance Techniques program are: to define and develop intelligent security auditing techniques; to develop a systematic approach to penetration analysis and countermeasures; and to properly apply Artificial Intelligence (AI) to computer security.

The Integrated Assurance Techniques program will effect an enhanced level of security to existing DoD computer systems by providing the ability to recognize and prevent unwanted system intrusion attempts. It will also provide the capability to include system penetration countermeasures during development of future DoD computer systems. The program contains four (4) projects (Intelligent Security Auditors, Generic Penetration Countermeasures, Self-assuring Techniques, and Intrusion Detection Expert Systems) each with multiple tasks (in- house and contractual).

C.5.7.6 National Security Agency

Title: Evaluation and Analysis Techniques
Objectives: The need for Government/Industry cooperation in information system security has led to industry development of widely marketable products meeting targeted levels of trust, as outlined in the DoD Trusted Computer Systems Evaluation Criteria (TCSEC). The objective of this program is to provide the tools necessary to support that national effort in terms of product evaluation, system assessment and certification, and techniques to improve future developments. This program also addresses the development of methods for a unified risk management life cycle strategy.
Approach/Thrusts: The program contains seven (7) projects (Evaluators' Tool Kit, Developers' Tool Kit, Automated Risk Management, Covert Channel Analysis Techniques, Tactical Computer Security (COMPUSEC) Certification and Accreditation, Risk Modeling, and Security Analysis of Higher Order Languages) each with multiple tasks (in-house and contractual).

C.5.8 Cryptographic Techniques

C.5.8.1 Navy: Office of Naval Research

Title: Machine Proof of Cryptographic Protocols

C.5.8.2 Office of the Assistant Secretary of Defense (Production & Logistics)

Title: Protection of Logistics' Unclassified/Sensitive Systems
Objective: Protection of unclassified but sensitive data in accordance with the Computer Security Act of 1987. Data includes defense-related technical orders, engineering data, acquisitions, and electronic fund transfers over communications networks and transportable storage media.
Approach/Thrusts:

- Sponsor development of Public Key Encryption techniques with the National Institute of Standards and Technology and the National Security Agency (NSA).
- Develop digital signature formats and procedures with NIST, DLA, industry, and OSD procurement.
- Perform laboratory demonstrations using common DoD and industry network configurations.

Payoffs:

- Data security for business applications
- Unclassified crypto key management usable by business
- Feasible, standard digital signature methodology for documents.

Major Users and Related Activities: The project is receiving technical guidance from NIST and NSA

C.6 Simulation and Modeling Technology

C.6.1 Air Force: Rome Air Development Center

Title: Distributed System Modeling Environment

Objective: Develop an environment which supports the simulation of distributed systems, exploiting the commonality of simulations, and providing a coherent set of data collection and analysis tools.

Approach/Thrusts: Perform a detailed analysis and top-level design of an overall architecture to implement the objective, as well as specific implementation of some components of that design, including the user-interface, the interface between the user-interface and the distributed system modeling environment, and a populated database of objects for conducting an evaluation using the Simulation Driver Integration software as the System Under Study.

Payoff: Standardization of the interfaces and tools for the simulation user.

Major FY89 Accomplishments: ISM was housed on the RADC Distributed System Branch's Digital Equipment Corporation (DEC) computer system using the VMS operating system.

Major Users and Related Activities: The end customer is DoD.

C.6.2 Air Force: Rome Air Development Center

Title: Object Oriented Battlefield Simulation Development

Objectives:

- Develop an object-oriented battlefield simulation to support integration of stand-alone decision aids.
- Develop a prototype scenario generation facility to support the complete development cycle of command and control support systems.
- Assess new technology advancements in knowledge-based simulation.

Approach/Thrusts: As the battlefield becomes more complicated due to advances in sensors, electronics, and weaponry, it becomes more difficult to accurately model and simulate. These factors, along with the need for simulations to be more flexible, interactive, and intelligent are the motivating forces for this effort.

This work consists of extending a prototype simulation developed inhouse at RADC to serve as a research vehicle into extending and improving the capabilities of conventional simulation technology. The extensions will utilize existing research performed at RADC which has focused on building both an object-oriented simulation language (ERIC), and Map Display System. Work will leverage off an air-based simulation developed inhouse to include both the ground force models and a scenario generation facility. A facility of this nature is unique to military simulations and shows great promise in reducing the enormous time associated with current scenario building procedures.

Payoff:

- Flexible, interactive, and intelligent simulation for use in both training and battle management testbed environment.
- Experimental scenario generation prototype aimed at reducing long lead times associated with scenario development.
- Assessment of new simulation technologies.

Major FY89 Accomplishments: Contract began late in FY89 (Aug.). Time spent in FY89 consisted of contractor getting familiar with government furnished software.

Major Users and Related Activities: This project is targeted at the next generation simulation environment. It will also be used to help drive a major effort dealing with the development of a testbed environment for testing out expert systems *aimed at supporting the* C^3*I environment.*

This project is directly related to RADC's inhouse simulation work. The effort and inhouse work are being performed in a cooperative manner.

C.6.3 Air Force: Rome Air Development Center/Strategic Defense Initiative

Title: Internetted System Model (ISM)

Objective: Develop modeling tools to support the analysis and system design of internetted heterogeneous computer networks appropriate to the SDI environment.

Approach/Thrusts: To extend existing tools to consider the traffic flow and system functionality associated with multi cluster, internetted configurations of ground, airborne, and satellite nodes.

Payoff: ISM has a user-friendly interface which permits the network analyst to enter parameters which describe the simulated system under study, invoke the model, and produce output statistical reports and graphs. ISM is oriented for the network analyst who is a non-programmer. It has a user-interface that requires minimal training time to use. The user-interface permits on-line help and error messages. All the input parameters are saved in libraries for later use.

Major FY89 Accomplishments: ISM was hosted on the RADC Distributed System Branch's Digital Equipment Corporation (DEC) VAX 8650 computer system using the VMS operating system.

Major Users and Related Activities: The end customer is DoD. Harris Corporation has a copy of the software for potential application on a NASA contract. MITRE has requested a copy.

C.7 Management Support

C.7.1 Army: AIRMICS

Title: DY10-05/Decision Support Systems

Objective: Develop techniques and methods to improve the quantity and quality of information to support decision making.

Approach/Thrusts:

- Individual Support
- Executive Support
- Group Support
- Expert Support

Payoff:

- Improve decision-making quality
- Establishment of cost-effective methodologies for developing/delivering decision support tools

Major FY89 Accomplishments:

- Completed FORSCOM DSS Study
- Completed Beta test
- Completed management guidelines for expert system development for Mobilization Planning Expert System.

C.7.2 Army: AIRMICS

Title: DY10-07/Management of Information

Objective: Develop concepts to support the use of technology in the management and operations of information intensive segments of the Army.

Approach/Thrusts:

- Information Centers
- Information Systems Transition Planning
- Centers/Consortium
- Future Architectures

Payoff:

- Effective/efficient technology insertion
- Information Mission Area integration
- Aids in recruitment
- Provides mechanism to increase research at historically black colleges and universities

Major FY89 Accomplishments:

- Information Architecture Reference Model (Report)
- Completed Integrated Information Mission Area Information Center Guidebook (report)
- Established pilot Video Tele Conferencing network at historically black colleges and universities and Army sites
- Participation in NSF Centers (Lehigh and Georgia Tech/U. of Arizona

C.7.3 Air Force: Wright Research and Development Center

Title: Hypermedia Avionics Software Support

Objective: Develop and demonstrate a Hypermedia based Avionics Software Support Capability.

Approach/Thrust:

- Select an avionics operation Flight Program for support demonstration.
- Determine requirements for a hypermedia system.
- Develop and demonstrate the support system.

Payoffs:

- Simple, natural style of information access.
  - — No key words/phrases to remember.
  - — No queries to compose.
- User-controlled level of detail presented.
- Information access time decreased.
- Documentation update, maintenance, and distribution time reduced.

C.7.4 OASD (P&L/Systems)

Title: Computer Aid for Acquisition and Logistics Systems (CALS)

Objectives: Reduce cost and increase responsiveness and readiness through automation of government and industry interaction and product delivery throughout the system life cycle.

Approach/Thrusts:

- Identify key standards and technologies needed to automate support for the DoD system life cycle.
- Promote sharing of information during acquisition and development through the Integrated Weapon Systems Data Base; a distributed data base residing at DoD and industry locations.
- Form industry / Dod working groups to address specific issues and formulate standards and procedures.
- Insert CALS standards into the national standards development process through support for NIST and ANSI committee efforts.
- Publish CALS guidance and strategies and encourage industry conferences and other forms considering standardization areas.

Payoff:

- Reduced cost to DoD and industry to develop and maintain technical products.
- More timely DoD and industry interaction for developing systems.
- Forum for industry feedback considering automation of certain acquisition/technical procedures.

Major FY89 Accomplishments:

- Adoption of the Product Description Exchange Specification including automation of software specifications.
- Initiation of the CALS Integrated Technical Information Services concept which addresses significant systems and management issues related to Integrated Weapon Systems Data Base implementation.

C.7.5 OASD (P&L)/EDI

Title: Electronic Data Interchange

Objectives: To automate common DoD business processes using emerging standards.

Approach/Thrust:

- Participate with NIST and ANSI committees in development of standards and procedures foı EDI.
- Implement emerging standards at Lawrence Livermore Labs and connect using a variety of communications facilities with DoD and industry facilities
- Determine strategies for ensuring interoperability.
- Use EDI platform to test interoperability of various DoD and industry implementations and develop methods to use automation to address significant DoD problems, e.g. prompt payment.

Payoffs:

- Reduced cost and increased productivity for DoD and industry.
- Options identified for streamlining acquisition process.
- Reduced cost to DoD agencies implementing EDI
- Stable technical environment for the development of DoD policy for EDI and related contracting and technical documentation areas.
- Identify common systems elements and processes requiring DoD management, e.g. vendor validation and interoperability testing.

C.7.6 Defense Logistics Agency

Title: Document Capture and Management

Objective: Investigate options to capture all incoming and outgoing documents (paper and electronic) to facilitate integrated retrieval, tracking, and coordination.

Approach/Thrusts:

- Integrate hardware and software available off existing contracts.
- Build a sample storage data base with search and retrieval software for documents in a combination of formats (scanned image, ASCII, SGML).
- Investigate tracking alternatives which facilitate coordination between offices and all for revision levels, annotations, and comments.
- Investigate search capabilities using human language input considering typical problems (e.g. synonyms, homonyms, contractions, typographical errors).

Payoffs:

- Reduce the costs of paperwork production, handling, and tracking.
- Increase the efficiency of office by reducing paper processing time.
- Maintain managerial control of document flow and tasking.

C.7.7 Defense Logistics Agency

Title: Voice Recognition System for Warehouse Management

Objective: Investigate the use of human speech to automate the entry of warehouse control information.

Approach/Thrusts:

- Capture information in the warehouse using both interactive voice terminal and cassette tape recorders.
- Process the human speech using an adaptive AI system to recognize a limited vocabulary and generate equivalent ASCII data.
- Build a prototype system in a warehouse to test the concept.

Payoffs:

- Reduction of costs for data entry, especially where the use of computer keyboards would be awkward.
- Augment the use of bar code scanning where impractical and for the capture of other types of date (e.g. vendor IDs, inventory counts).
- Support spoken status queries where terminals are not practical.

C.8 Education

C.8.1 Navy: Office of Naval Research

Title: Computer Technology Status Report

Objectives:

- Provide a 6.2 focus for the next generation computer resources program (NGCR)
- Stimulate university research, national initiatives and corporate IR&D in research areas directly feeding NGCR
- Make use of the open systems concept of NGCR to continually transition new computer technology more rapidly into Navy systems

Approach/Thrusts:

- Computer technology and its potential impact on Navy systems is not always well understood, particularly by those not conversant with the jargon. In addition, the technology is moving so rapidly that it is difficult to forecast the future with any certainty. A report and briefing for senior Naval officials will be developed and circulated in an attempt to improve understanding and support

Payoff:

- Evolutionary transition of most current computer technology to early Navy use
- Minimal Navy investment in national interest area with maximum results

Major Users & Related Activities:

- This project is directly targeted at NGCR. Membership on the NGCR committees assures close cooperation and well defined research objectives
- This project will also interface with the JIAWG through NADC
- The output of 6.1 programs in ONR and DARPA are being closely monitored. Primary programs of interest are ONR hard realtime systems in Ada and DARPA ESKIT

C.8.2 Navy: Naval Research Laboratory

Title: Intelligent Tutoring

Objectives: Improve effectiveness of technical training through use of adaptive computer-based instruction.

Approach/Thrusts:

- Develop an authoring system for navigating in a knowledge and exercise decision tree
- Develop techniques for translation of English-like rules into machine executable form
- Demonstrate feasibility in a complex subject matter training situation

Payoff:

- More effective training and better performance measurement
- Reduced training costs
- Effective methods for incorporation of embedded training in deployed systems

Major FY89 Accomplishments:

- Developed prototype learning environment
- Began incorporation of heuristic guidance

C.9 Computing Facilities

C.9.1 Army: CECOM

Title: Software Engineering Facility and Interns (A094 Tactical Software Engineering Technology)

Objective: Provide a state-of-the-art computer resources network to support the in-house program at CECOM's Center for Software Engineering and provide a one-year, intensive hands-on experience for second year AMC interns exposing them to real-life problems and provide formal training in software engineering.

Approach/Thrusts:

- Utilize standards to maximum extent possible
- Provide periodic backups and technical support to projects
- Provide 3 interlocked networks: Administrative Network, Engineering/Assessment Network, and R&D Network
- UNIX-based systems augmented by other operating system, as necessary
- Assign interns to on-going in-house software engineering projects
- Arrange for series of software seminars to be given to the interns & to the technical staff
- Establish contractual relationship with Monmouth College for formal graduate training in software engineering

Major FY89 Accomplishments:

- Awarded five year contract to Monmouth College for formal post-graduate training is software engineering
- First class of 21 interns completed training and were assigned permanent positions within AMC
- Second class of 32 interns currently completing their training
- Third class expected to arrive January 90
- Established CECOM/Industry Documentation Task Force

Payoff:

- A facility that promotes both research & experimentation, and administrative functions to occur simultaneously
- A facility that is flexible to changes
- An operational facility that supports program projects
- Provide trained software engineering personnel that will be productive Government employees
- Overcome the shortage of qualified software personnel within the Government

C.9.2 Navy: Naval Ocean Systems Center

Title: Networking and Infrastructure: High performance computing project

Objectives:

- Provide a vehicle for Navy participation in high performance computing national initiatives exemplified by "A Research and Development Strategy for High Perfoımance Computing" and "A National Computing Initiative: The Agenda for Leadership"
- Achieve early transition of major research breakthroughs in high performance computing to critical Navy applications

Approach/Thrusts:

- "A Research and Development Strategy for High Performance Computing" identified networking as one of the four primary research areas. This includes researcher interaction, creation of a cadre of trained researchers, creation of network/shared facilities, and formal professional society participation. This task was to fund this activity for Navy laboratories.
- Funds were deleted prior to completion of any significant efforts

Payoff:

- Networking will increase productivity through cooperation
- Direct transition to three critical Navy systems (OSS, HIGAIN, FDS)

Major Users and Related Activities:

- The HPC project is being coordinated with DARPA, NSF, DOE and other national initiatives in high performance computing
- There are three Navy projects currently targeted for transition of HPC products, Operation Support System (OSS), Fixed Distributed System (FDS), and the High Gain Initiative

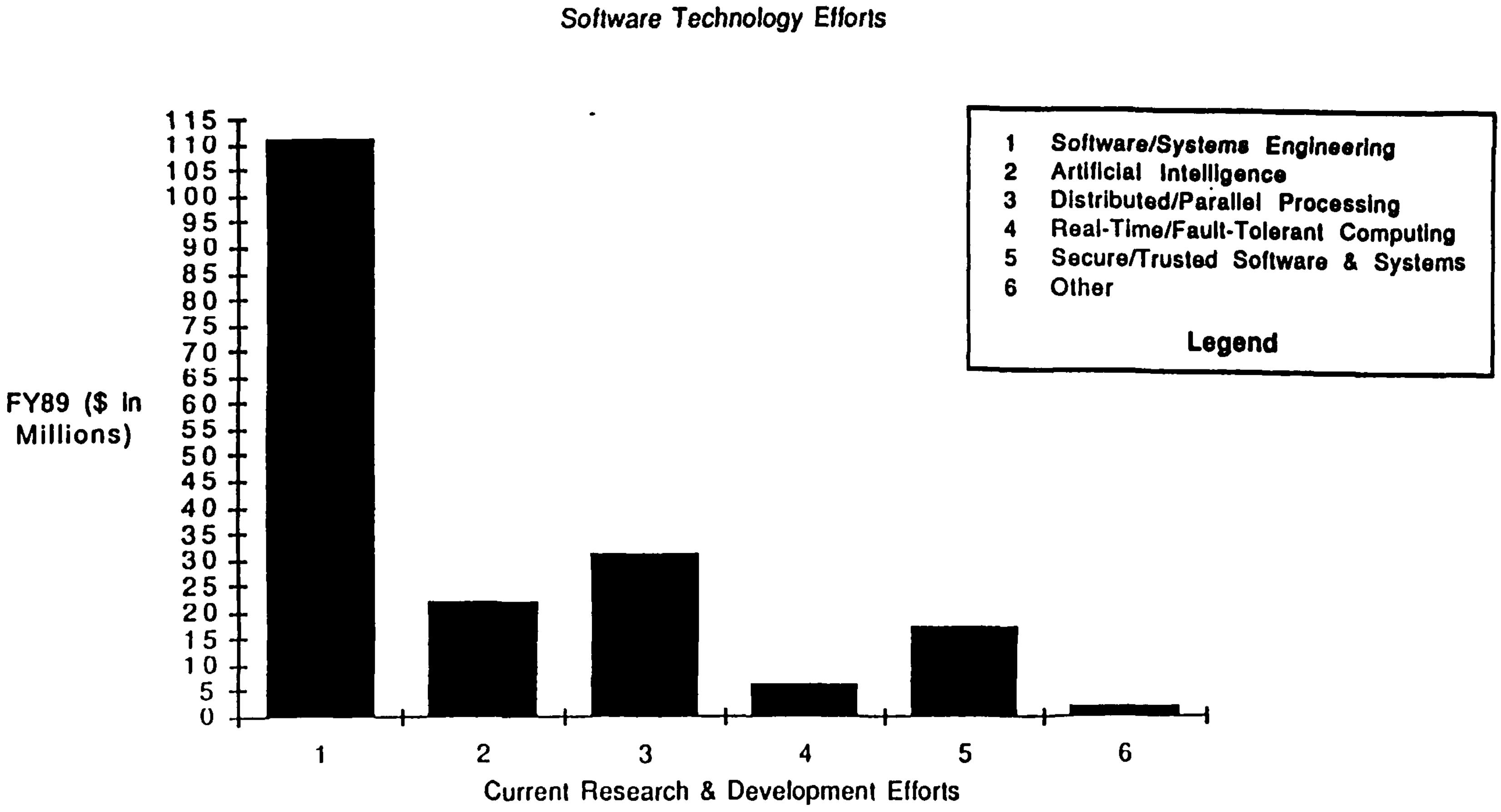
Software Technology Efforts
FY89 ($ In Millions)
115
110
105
100
95
90
85
80
75
70
65
60
55
50
45
40
35
30
25
20
15
10
5
0
1
2
3
4
5
6
Current Research & Development Efforts
1 Software/Systems Engineering
2 Artificial Intelligence
3 Distributed/Parallel Processing
4 Real-Time/Fault-Tolerant Computing
5 Secure/Trusted Software & Systems
6 Other
Legend

Figure C-1

ANNEX D

Cross-Cutting Issues

February 9, 1990

CONTENTS

ANNEX D

D. Cross-Cutting Issues

The means available to the DoD to effect the process and characteristics of software include management, policy, personnel, and technology. However, there are several highly visible and critical issues that must be addressed across these areas. They include the software process, software reuse, high assurance and secure/trusted software, real-time software, and parallel/distributed software. This annex provides a high level review as background and motivation for the required actions of Volume I.

D.1 Software Process

The software process embraces many activities, including requirements engineering, costing and forecasting, validation, design, implementation, verification, test and evaluation, documentation, adaptation, upgrade and corrective maintenance. Each of these activities is itself critical to the overall software life cycle, and is the subject of attention in this DoD Software Master Plan, with attention given to both management and technical issues.

A major source of leverage, however, comes not from a consideration of any single one of these activities, but rather from a consideration of the ways they link together into the overall software process associated with a system life cycle, from initial conceptualization to system retirement. Similarly, development of effective automation support for this software process requires an integrated focus.

Development of a "precedented" system, whose architecture and general characteristics are well-known to managers and developers as a result of successful prior experience, can generally be conducted in a phased manner, following a traditional "waterfall" plan of activities. The waterfall model can permit errors and problems from one stage to propagate to the next, with consequent impact to schedule and cost as the rework workload increases. Within this model, focus on obtaining defect-free quality can reduce downstream costs and risks significantly.

Developments in which there are risks associated with requirements specification, design, test and evaluation, or specific required product characteristics, must be treated differently. "Unprecedented" developments require process models that provide for explicit risk reduction and management through iteration, prototyping, early validation, and other means. Risk reduction models for such systems are often called "iterative" or "incremental" models.

Codification of complete requirement specifications at development outset without means for validation can create very high downstream risks. Requirements often change as systems are developed and throughout a system lifetime. For this reason, iterative refinement of requirements with means for early validation and commitment, should be enabled. Consequently, when required, the acquisition process should recognize and encourage integrating requirements engineering steps with early development and prototyping.

Early participation by users, Post Deployment Software Support (PDSS) organizations, and test and evaluation agencies provides a means to reduce risks and increase confidence in early validation results. Direct interaction with these parties should be encouraged to ensure the success of the overall software life cycle.

PDSS is software redevelopment required to change deployed software for purposes of correcting existing (but often not discovered during development) errors and enhancing the software to provide new, additional capabilities. PDSS is one of the least understood and highest cost aspects of the life cycle of a DoD software sensitive system. Current experience shows that 60 to 70 percent of software costs are incurred during this phase of the life cycle. In this era of declining defense budgets, it is imperative that actions be taken to reduce PDSS costs, while increasing its effectiveness. These somewhat contradictory requirements, in turn, necessitate a fundamental change in current system and software acquisition practices.

D.2 Software Reuse

Software reuse includes the reuse of software designs, systems architectures, software components, documentation, concepts and source and object code. It is seen as a technique to mitigate many of the problems currently faced by the DoD in its software sensitive systems. Reuse leverages the best solutions, designs, and architectures for use in defense systems. The skills of DoD's best software engineers can thus be maximized and the efforts utilized in multiple projects. Reliability and assurance measures associated with software are expected to increase as more software objects are reused and experience with reuse is gained. Maintenance costs are expected to decrease as high-assurance reusable components begin to comprise a larger part of DoD systems.

To achieve such benefits, an infrastructure conducive to and supportive of software reuse must be developed in the defense community. In particular, the DoD must remove current disincentives associated with software reuse and must provide economic incentives to develop reusable software objects. In addition, application domains must be analyzed and standard interfaces must be developed to facilitate software reuse. Software repositories are needed to support the reuse of software systems, subsystems, components or routines, designs, architectures, and documentation. These repositories must support extraction of objects and integration of objects with existing and planned DoD systems Such repositories must provide a mechanism for the operational and maintenance support of reuse objects, including provisions for reporting deficiencies, problems, correcting errors, improving documentation, and modifying or enhancing the reuse object. Finally, techniques must be established to validate reusable objects: i.e. to prove that such objects perform as expected and to ensure that sucl objects meet applicable DoD standards and regulations.

D.3 High Assurance and Secure/Trusted Software

High assurance software is that software which requires demonstrated performance in terms of quality attributes, e.g., reliability, availability and maintainability, and safety, in addition to code correctness. Secure/trusted systems are a kind of high assurance system in which security properties must be established with high levels of confidence. Security properties include confidentiality, integrity, and access control. To achieve high assurance and secure/trusted software there must be: (1) an appropriate (often formal) articulation of realistic requirements, (2) the availability of software engineering techniques for developing software with the desired attributes that scale to the development of large systems, (3) a cost effective process to quantitatively demonstrate achievement of the required attributes, at both the software and system levels; and (4) a validated evaluation and assessment process that not only provides the basis for current performance, but also a predictive basis for system success or failure.

The issues of high assurance and secure/trusted software must be addressed through a variety of mechanisms. Since the techniques for specifying and developing high assurance and secure/trusted software, particularly in a distributed environment of shared resources, are a relatively new area of research, there are many generic facets that still require basic and applied research. Such factors include the following: how to support the development of high quality code; how to avoid the insertion of malicious code; how to detect and eliminate intentional and unintentional errors; and how to prove that properties such as correctness, reliability, or integrity exist in the code. Formal, scalable techniques must be developed to support these needs.

D.4 Real-Time Software

One of the most important categories of software applications enabled by modern computers, particularly but not exclusively for defense, is the class of applications with real-time constraints. In this context, real-time denotes that successful performance in a particular application depends on satisfying (usually complex) timing properties, such as response deadlines. Real-time systems are characterized by the need to be able to control, respond to, or interact with external environmental phenomena, such as object motion, temperature or pressure, concentration of mixed ingredients, physical position, or other digital and analog systems. Such systems must react responsively to these external stimuli, and most importantly, they must behave in a predictable way. Frequently such systems are called upon to operate in situations where failure to satisfy timing constraints results in immediate catastrophic events, with little or no opportunity for human intervention.

In spite of the widespread acceptance and exploitation of real-time computing technology in both civilian and military contexts, it is not broadly recognized nor appreciated that real-time computing is not a well-understood discipline, and that it is currently fraught with serious limitations in its ability to be used with credibility. The sorts of applications that high-performance computing engines have made conceptually possible are far beyond the state of the art that real-time software can currently support. Indeed, even modest applications quickly lead to intractable problems in verifying that real-time software truly accomplishes the goals established in a system's specification.

The most fundamental research issues for real-time software are (1) how to formally express and verify real-time system specifications, designs, and implementations; and (2) how to allocate and schedule real-time system resources effectively and predictably under broad operating conditions. These two central issues demand sustained, energetic, creative basic research of the sort normally found in universities. There is an absolute need to invest in basic research on proof techniques, program semantics, scheduling theory, dependability, state-space modeling, and other approaches that offer the promise of being able to create a scientific basis for real-time system specification and construction.

Even before the hoped for breakthrough techniques are discovered in basic research, industry, together with the Government's applied research establishment, can also play an important role in producing higher-quality real-time systems by being much more active in technology transition. There are many partial results in the real-time computing literature that have been discovered in the past fifteen years, particularly in scheduling theory. By and large, industry does not make use of these results to any great degree, preferring to use techniques handed down from previous generations of simpler real-time systems in the belief that "risk" is mitigated in doing so.

Finally, the Government itself must become a much more informed buyer of projects that include real-time computing systems. It must determine whether certain desired applications can be achieved with current and projected understanding of real-time computing. (The answer will sometimes be "no".) Government program managers must take explicit steps to promote the use of the most advanced techniques possible, and to continually assess whether such techniques are being applied. The program manager must be assisted in this responsibility by establishing formal minimum standards and guidelines for real-time systems construction.

D.5 Parallel and Distributed System Software

Distributed systems have been in use in the DoD for many years, in configurations ranging from embedded systems to heterogeneous networks of workstations. Despite the broad range of uses, distributed computing support has generally been ad hoc, especially in embedded systems. There are few standard components or even architectures for managing distributed system operation in these settings. More recently, parallel systems have begun to emerge in the marketplace with the promise of providing high performance computing in a cost-effective and scalable manner. This is already beginning to have impact in the AIS community (e.g. as a source of low cost database transaction cycles), in the MCCR community (e.g. as a means to embed high performance sensor processing systems), and in the scientific/engineering computing community (e.g. in the form of next generation supercomputers). Scalability implies that systems can be scaled up in capacity as computational requirements evolve and as computing modules become faster and more compact.

Parallel and distributed systems pose challenges in a number of software areas. Standard systems software bases must be developed that are appropriate to the various computing domains. There is strong evidence that there are simple models of concurrent computation that can be applied in wide varieties of parallel configurations.

Adoption strategies for high performance computing systems must be developed that balance the cost-effectiveness and scalability with risks. Risks can be reduced through early investment in the software aspects of the system; i.e. "software first" and prototyping may be useful approaches to take in developing solutions to high performance computing problems.

Technology base efforts are currently addressing issues related to algorithms, programming language, verification and testing, and other areas. It must be recognized in requirements formulation, however, that the parallel systems offer new opportunities. For example, multiprocessors enable runtime auditing of computations without performance penalty. Also, highly parallel systems offer new computational approaches to classical scientific/engineering problems, enabling, for example, replacement of approximate analytical techniques with simulations.

Distributed systems technology has enabled cost-effective scalable computation-based battle simulations. The acquisition of such systems requires the establishment of standards to enable incremental improvements to existing systems and to enable competition among multiple vendors for additional function and capacity.

ANNEX E

Review of Past Studies on Software Issues

February 9, 1990

CONTENTS

ANNEX E

E. Review of Past Studies on Software Issues

There have been numerous studies and reports over the years that have addressed the issues involved in developing and maintaining software for the DoD. Many of these documents provided recommendations to the DoD for ameliorating areas where problems were cited. The DoD was able to respond and implement many of the specific recommendations. Quite often, however, such recommendations were of a global nature or were unspecific and, therefore, difficult to implement directly.

This annex identifies existing software issues*, maps the recommendations of twelve of the most comprehensive previous studies to these issues, and identifies actions that have been taken by DoD with respect to the recommendations of each issue area. It should be noted that some of the DoD actions taken may not have been in direct response to a specific recommendation, but rather in response to some alternative driving factor. In addition, the effectiveness of an individual action in addressing all aspects of a corresponding issue varies.

Table E-1 lists the selected studies. Section E.1 provides the outline of software issues. Section E.2 provides the full text of the recommendations from each study. Section E.3 provides a list of actions taken thus far within the DoD. Finally, the information is summarized in Table E.2 which identifies each of the recommendations related to a particular software issue, along with the actions that have been taken that correspond to that same issue. The section labels in Table E.2 correspond to the divisions found in Section E.1. The recommendations and actions in Table E.2 refer back to the full text of each given in Sections E:2 and E.3.

This annex, which represents a consolidated perspective of the current status of related DoD activities, also provides the most comprehensive assessment of the DoD response to the specific studies. As such, the information was instrumental in the formulation of specific actions still required within the DoD to address specific software issues.

E.1 Outline of Software Issues

E.1.1 Policy

E.1.1.1 Lack of a defined overall software management, development, and requirements policy

E.1.1.2 Policies not current with or conducive to latest technology

E.1.1.3 Lack of policy to support reuse

E.1.1.4 Inadequate data rights policy

E.1.1.5 Inadequate PDSS and software maintenance policy

E.1.1.6 Inadequate enforcement of policies

E.1.1.7 Software policies uncoordinated and in conflict with one another

E.1.1.8 Lack of appropriate standards

E.1.1.9 No national strategy for the software engineering field

* The original basis of this list of issues/problems was a list created at the SEI Software Problems Workshop of October, 1988 [SEI88b] It has since been significantly modified based on issues identified in additional studies and reports.

Table E-1: Major Studies on the DoD Software Problem

Report	Date	Reference
Report of the Defense Science Board Study Panel on Technology Base	1981	[DSB81]
Report of the DoD Joint Service Task Force on Software Problems	1982	[DOD82]
The High Cost and Risk of Mission-Critical Software, USAF Scientific Advisory Board Report	1983	[USAF83]
DoD Management of Mission Critical Computer Resources, Council of Defense and Space Industry Associations Report	1984	[CODSIA84]
Report of the Defense Science Board Task Force on Military Software	1987	[DSB87]
Ada Board Response to the Report of the Defense Science Board Task Force on Military Software, Ada Board Report	1988	[BOARD88]
Summary Report on the Defense-Wide Audit of Support for Tactical Software, Office of the Inspector General	1988	[OIG88]
Proceedings of the Workshop on Executive Software Issues, Software Engineering Institute	1988	[SEI88]
Report of the Workshop on Military Software	1988	[ZRAKET88]
Report of the Army Materiel Command Software Task Force	1989	[AMC89]
Software Technology Development and Deployment Plan for the DoD Technology Base, Institute for Defense Analyses	1989	[IDA89]
Adapting Software Development Policies to Modern Technology Air Force Studies Board	1989.	[AFSB89]

E.1.2 Management

E.1.2.1 Management Methods

E.1.2.1.1 Inadequate software management concepts, methods, practices

E.1.2.1.2 Lack of management attention/commitment to software issues

E.1.2.1.3 Lack of commitment to Ada

E.1.2.1.4 Lack of adequate planning and focus on front-end processes of life-cycle

E.1.2.1.5 Inadequate risk assessment

E.1.2.1.6 Poor management methods for software support

E.1.2.1.7 Software planning not integrated with system planning

E.1.2.1.8 Inadequate metrics and monitoring tools

E.1.2.2 DoD Management Structure

E.1.2.2.1 Efforts are uncoordinated

E.1.2.2.2 Lack of clearly defined roles and responsibilities

E.1.2.2.3 Structure is inadequate

E.1.2.2.4 No structural link between deployed software needs and the software Research & Development community

E.1.3 Life Cycle Management

E.1.3.1 Requirements

E.1.3.1.1 Requirements are ill-defined, incorrect, and/or do not meet user's needs

E.1.3.1.2 Requirements undergo significant uncontrolled change

E.1.3.1.3 Requirements lack system view

E.1.3.1.4 Inadequate documentation and allocation/traceability o^{c} requirements

E.1.3.1.5 No capability to perform cost/benefit analysis of requirements change

E.1.3.2 Costing/Forecasting

E.1.3.2.1 Lack of resource forecasting model for new methods/technologies

E.1.3.2.2 Lack of historical basis for predicting/estimating software

E.1.3.3 Acquisition

E.1.3.3.1 Software capabilities not adequately addressed in system contracting

E.1.3.3.2 Current acquisition process inhibits innovative methods

E.1.3.3.3 Difficult under current contracting mechanisms for contractors to capitalize software assets

E.1.3.3.4 Inadequate education with respect to innovative acquisition options

E.1.3.3.5 Lack of productivity and quality incentives

E.1.3.4 Design/Analysis

E.1.3.4.1 Software product designs unwieldy for complex, high performance, and changing requirements

E.1.3.4.2 Mechanism lacking for early collaboration to establish and resolve common interfaces

E.1.3.5 Development Methodology

E.1.3.5.1 Waterfall model is usually inadequate, but alternative models are not fully developed and must address complex DoD application domain

E.1.3.5.2 Methodologies for evaluating development methods generally do not exist

E.1.3.6 Languages

E.1.3.6.1 Inability to calibrate tradeoffs of language performance vs. downstream system capabilities

E.1.3.6.2 Need to integrate 4th/5th generation languages into DoD mainstream

E.1.3.6.3 Ability to obtain predictable real-time performance with Ada has not been demonstrated

E.1.3.6.4 No widespread implementation of Ada Chapter 13

E.1.3.7 Development and Support Tools/Environments

E.1.3.7.1 Insufficient tech base hampers ability to use tools

E.1.3.7.2 Lack of consensus on the use of methods and tools results in their inconsistent application

E.1.3.7.3 Tools do not support the full life cycle

E.1.3.7.4 Tools do not integrate or port

E.1.3.7.5 Lack of tools that support parallel and distributed systems, particularly runtime instrumentation

E.1.3.7.6 DoD has a heavy bias toward custom-built methods, tools, and software

E.1.3.7.7 Adequate support environment must be planned for and acquired during software development

E.1.3.7.8 Ineffective software configuration management tools

E.1.3.7.9 Inadequate tools to collect data needed for software measurement

E.1.3.8 Test and Evaluation

E.1.3.8.1 Inadequate verifiability, suitability, and predictive capability of existing testing methods

E.1.3.8.2 Need to fully integrate testing and evaluation into development activities

E.1.3.8.3 Failure to apply existing methods

E.1.3.8.4 Product assurance groups lack leverage

E.1.3.9 Adaptive, Corrective, and Preventive Maintenance

E.1.3.9.1 No ability to develop complex software that operates correctly and is maintainable

E.1.3.9.2 Software complexity tends to grow during development and with evolutionary upgrades

E.1.3.9.3 No link between developers and maintainers

E.1.3.9.4 Lack of technology for support of deployed systems, including support of low quality software

E.1.3.10 Modernization/Upgrades

E.1.3.10.1 Lack of techniques for developing evolvable systems

E.1.4 Product

E.1.4.1 Fulfillment of Requirements

E.1.4.1.1 Lack of generally accepted and understood acceptance criteria

E.1.4.1.2 System not interoperable

E.1.4.1.3 Quality of software degrades disproportionately with the size and complexity of software

E.1.4.2 Reliability/Fault Tolerance

E.1.4.2.1 Systems are not reliable and do not recover adequately from processor faults

E.1.4.2.2 There is not an adequate concept of and theory of software reliability

E.1.4.3 Documentation

E.1.4.3.1 Software document is not prepared in a standardized, easily usable manner

E.1.4.3.2 Documentation is incomplete or poor quality

E.1.4.3.3 Tailored documentation needs not addressed

E.1.4.3.4 Contractor proprietary rights in conflict with government support documentation needs

E.1.4.4 Adaptability to New Technologies

E.1.4.4.1 Current systems not evolvable or transportable

E.1.4.5 Quality Measurement

E.1.4.5.1 Product quality metrics are deficient

E.1.4.5.2 Causal relationships among program management process, software development process, and product quality are not proven

E.1.5 Software Reuse and Commercial Off-The-Shelf (COTS) Software

E.1.5.1 Technology

E.1.5.1.1 There is no adequate concept or methodology for software reuse

E.1.5.2 Incentives

E.1.5.2.1 Incentives and policies are lacking to encourage use of COTS software & reuse

E.1.5:3 Access/Retrieval

E.1.5.3.1 There are no domain-specific libraries for reusable components

E.1.5.3.2 There are inadequate means to collaborate with commercial industry to develop COTS components that address defense requirements

E.1.6 Personnel

E.1.6.1 Management Personnel

E.1.6.1.1 Upper management lack of familiarity with software

E.1.6.1.2 Software Managers lack expertise/education in software engineering

E.1.6.1.3 Poor utilization and allocation of software professionals

E.1.6.2 Acquisition Expertise

E.1.6.2.1 Acquisition people have inadequate knowledge of software

E.1.6.3 Real World/Large-Scale System Expertise

E.1.6.3.1 Academic software engineering courses don't provide adequate experience with large-scale systems

E.1.6.4 Ongoing and Refresher Training

E.1.6.4.1 Failure to educate for changing and/or immature technology

E.1.6.4.2 Need to improve individual productivity

E.1.6.5 Retention/Recruitment

E.1.6.5.1 Shortage of people with specialized skills

E.1.6.5.2 Best people go into industry

E.1.6.5.3 Lack of career paths

E.1.7 Technology Transition

E.1.7.1 There is no strategy for dealing with rapidly changing technology and engineering practice

E.1.7.2 There is a lack of awareness of existing technology

E.1.7.3 Need for technology demonstration projects

E.1.8 Security

E.1.8.1 Software products can not meet security requirements

E.1.8.2 Technology for secure distributed systems and new evolving architectures not available

E.1.8.3 Security capabilities are very costly

E.1.9 Technology Base

E.1.9.1 University research base in need of considerable enhancement in the areas of faculty, equipment, facilities, and support

E.1.9.2 Key technology areas lack adequate research funding

E.2 Recommendations from Reports

As much as possible, the actual text of recommendations from these reports is repeated below. The reports themselves are listed chronologically.

E.2.1 Report of the Defense Science Board Summer Study Panel on Technology Base, 1981 [DSB81]

1. Formulate vertically integrated technology base programs with fenced funding in Machine Intelligence and Advanced Software/Fast Algorithms.
2. Develop and use an investment strategy method to ensure linkage between investment decisions and technical needs, opportunities, and risks.
3. Increase university support for research and revise current procurement policies and regulations for universities.
4. Provide graduate fellowships with a requirement to work in DoD labs one year for each year of fellowship support.
5. Upgrade computer resources of universities

E.2.2 Report of the DoD Joint Service Task Force on Software Problems, 1982 [DOD82]

1. Refine and support software acquisition and support management practices
   a. Develop better monitoring approaches and tools
   b. Develop life-cycle model reflecting continuing changes in requirements.
   c. Develop contracting incentives for quality software and reuse
   d. Develop microprocessor/firmware policies
2. Stimulate technology research, development and utilization:
   a. Tools to assess impact of proposed changes to the system
   b. Design practices for systems with specific characteristics (highly parallel, distributed, reusable, etc.)
   c. Metrics for cost, productivity, and quality
   d. Integrated environment

   e. Documentation support
   f. Techniques for dealing with rapid technology changes
   g. Greater communication and coordination among Service elements
   h. Framework for technology evaluation and infusion
   i. Encourage IR&D in software
3. Develop and use expertise
   a. Define and advocate systems and software engineering career fields
   b. Encourage exchange of software personnel between government and industry
   c. Establish productivity measures

E.2.3 Report of the USAF Scientific Advisory Board, Ad Hoc Committee on the High Cost and Risk of Mission Critical Software, [USAF83]

1. Establish a focused, high-priority career path for software and computer system personnel.
2. Create a plan to evolve to a Deputy Chief of Staff level manager of USAF information resources, including mission-critical and embedded computers and software.
3. Establish a Software Engineering and Computer System Technology and Support Center to Collect and focus Air Force resources on software issues.
4. Establish an Air Force policy on software risk management.
5. Provide risk management information support.
6. Establish a policy on software oversight management.
7. Increase investment for software engineering tooling.
8. Increase funding for long range software technology research.
9. Acquisition practices must encourage technology transfer.
10. Increase use of Independent Verification and Validation (IV&V).
11. Investigate methodology to ensure high integrity software.
12. Upgrade Post Deployment Software Support (PDSS) personnel.
13. Standardize and increase capital spending for PDSS support environments.
14. Enable early participation by designated support organization.
15. Establish single management authority for assuring system integrity after deployment.

E.2.4 Council of Defense and Space Industry Associations Report on DoD Management of Mission-Critical Computer Resources, [CODSIA84]

1. DoD should optimize its use of the Nation's computer resources, changing from the current emphasis on equipment and instruction set architecture standardization to a higher-level standards approach with an emphasis on technology leadership.
2. DoD should modify existing policies as it transitions to the new policy:
   a. Update the Computer Resource Management directive (DoD Directive 5000.29).
   b. Issue the High Order Language directive (DoD Directive 5000.31) updated in June 1983.
   c. Control unproductive proliferation, but do not issue DoD Instruction 5000.5X.
   d. Allow the computer development programs now underway by the military services to continue, but with modifications to encourage technology insertion and more competition.
3. Develop an FFIT (Form-Fit-Interoperability-Transportability) computer resource management policy that will effectively achieve the goals outlined in this report.
4. Improve computer resource policy and initiative coordination.
5. Ensure systems compatibility, the use of Micro-Area, Local-Area, and Wide-Area networks, and architectural innovation.
6. Use Ada as a focus for technical activities necessary to achieve software transportability, reusability and interoperability.
7. Update hardware form and fit standards and current DoD instruction set architecture programs to focus in directions that will be compatible with FFIT policy.
8. Continue the Ada programming support environment effort and the Common Ada Programming Support Environment (APSE) Interface Set (CAIS) effort.

9. Expand run-time operating systems technology activities and develop common run-time operating system interface sets.
10. Integrate current initiatives - Ada, STARS, VHSIC, and Strategic Computing, and add a new *system*-level initiative.
11. Change standardization policy to raise the levels of standards, make them tailorable and support voluntary standards bodies.
12. Formalize and streamline computer resource management organizational structure at all levels within DoD.
13. Use DoD, industry, and academia resources in a team concept to establish computer resource technical and management working groups to address management and standardization issues.
14. Manage FFIT development to develop strategy and plans, to establish a new *systems* level initiative, to unify existing computer resource initiatives, to engineer and implement the "DoD computer resource standardization program plan," and to participate in industry's voluntary standards process.
15. Provide *fast* management oversight to survey and structure current practice, to implement interim policy, to transition to goal policy, and to improve instruction set architecture and product management.

E.2.5 Report of the Defense Science Board Task Force on Military Software, 1987 [DSB87]

1. Move STARS (to Electronic Systems Division) and develop a new plan with emphasis on early milestones
2. Task STARS, AJPO, SEI, SDI, and DARPA Strategic Computing Program to produce a one-time joint plan to demonstrate a coordinated DoD Software Technology Program.
3. Task the new STARS director to define a new set of program goals together with an implementation plan
4. Direct STARS to choose several real programs early in development and augment their funding to ensure the use of existing modern practices and tools.
5. Commit DoD management to a serious and determined push to Ada
6. Move the AJPO to the same organization as STARS and the SEI.
7. Keep the AJPO as the technical staff support for the DoD's executive agent
8. DoD policy should continue to forbid subsetting of the Ada language
9. DoD policy should increase investment in Ada practices education and training for both technical and management people.
10. Allow fourth-generation languages to be used where the full life-cycle cost-effectiveness of using the language measures more than tenfold over using a general-purpose language.
11. Focus a critical mass of software research effort on the software needs that are unique to the SDI objectives
12. Use evolutionary acquisition, including simulation and prototyping, to reduce risk
13. The Under Secretary of Defense (Acquisition) should adopt a four-category classification as the basis for acquisition policy: Standard (commercial off-the-shelf systems), Extended (Extension of current systems, both DoD and commercial) Embedded, (functionally unique and embedded in larger systems) Advanced, (Advanced and exploratory systems)
14. The Under Secretary of Defense should develop acquisition policy, procedures, and guidance for each category.
15. Under Secretary of Defense (Acquisition) and the Assistant Secretary of Defense (Comptroller) should direct Program Managers to assume that system software requirements can be met with off-the-shelf subsystems and components until it is proved that they are unique.
16. All the methodological efforts, especially STARS, should look to see how commercially available software tools can be selected and standardized for DoD needs.
17. DoD should devise increased productivity incentives for custom-built software contracts, and make such incentivized contracts the standard practice.

18. DoD should devise increased profit incentives on software quality.
19. DoD should develop metrics and measuring techniques for software quality and completeness, and incorporate these routinely in contracts.
20. DoD should develop metrics to measure implementation progress
21. DoD should examine and revise regulations to approach modern commercial practice insofar as practicable and appropriate.
22. DoD should follow the concepts of the proposed Federal Acquisition Regulation (FAR) 27.4 for data rights for military software, rather than those of the proposed DoD Supplement 27.4, or it should adopt a new "Rights in Software" Clause as recommended by Samuelson, Deasy, and Martin.
23. The Under Secretary of Defense (Acquisition) should update DoD Directive 5000.29 "Management of Computer Resources in Major Defense Systems", so that it mandates the iterative setting of specifications, the rapid prototyping of specified systems, and incremental development.
24. DOD-STD-2167 should be further revised to remove any remaining dependence upon the assumptions of the "waterfall" model and to institutionalize rapid prototyping and incremental development.
25. DoD Directive 5000.29 and DOD-STD-2167 should be revised or superseded by policy to mandate risk management techniques in software acquisition, as recommended in the 1983 USAF/Scientific Advisory Board Study.
26. Each Service should provide its software Product Development Division with the ability to do rapid prototyping in conjunction with users.
27. Each Service should provide its software Using Commands with facilities to do comprehensive operational testing and life-cycle evaluation of extensions and changes.
28. The Under Secretary of Defense (Acquisition) and the Assistant Secretary of Defense (Comptroller) should by directive spell out the role of Using Commands in the evolutionary and incremental development of software systems.
29. The Under Secretary of Defense (Acquisition) should develop economic incentives, to be incorporated into standard contracts, to allow contractors to profit from offering modules for reuse, even though built with DoD funds.
30. The Under Secretary of Defense (Acquisition) should develop economic incentives, to be incorporated into all cost-plus standard contracts, to encourage contractors to buy modules and use them rather than building new ones.
31. The Under Secretary of Defense (Acquisition) and Assistant Secretary of Defense (Comptroller) should direct Program Managers to identify in their programs those subsystems, components, and perhaps even modules, that may be expected to be acquired rather than built; and to reward such acquisition in the RFP's.
32. The Software Engineering Institute should establish a prototype module market, focussed originally on Ada modules and tools for Ada with the objective of spinning it off when commercially viable.
33. The Software Engineering Institute, in consultation with the Ada Joint Program Office, should establish standards of description for Ada modules to be offered through the Software Module Market.
34. Do not believe DoD can solve its skilled personnel shortage; plan how best to live with it and how to ameliorate it
35. Use DoD people for acquisition instead of construction
36. Establish mechanisms for tracking personnel skills and projecting personnel needs.
37. Structure some officer careers to build a cadre of technical managers with deep technical mastery and broad operational overview.
38. Enhance education for software personnel

E.2.6 Ada Board Response to the Report of the Defense Science Board Task Force on Military Software, 1988 [BOARD88]

1. The Ada Board recommends strongly that the Ada Joint Program Office (AJPO) continue to function at the OSD level.
2. The Ada Board recommends that coordination of Ada activities in accordance with DoD Directive 3405.2 be strongly enforced, and that the implementation plans include adequate provision for technical assistance to programs using Ada.

E.2.7 Summary Report on the Defense-Wide Audit of Support for Tactical Software, 1988 [OIG88]

1. Revise DoD Directive 5000.29 to include minimum requirements for a computer resource plan. The requirements should include: identification of software as configuration items, milestones and criteria for measurement of progress, use of software documentation standards, high-order programming language standards, software quality assurance standards, configuration management standards, software life cycle costs, specifications for contractor and DoD in-house support environments, provisions for transportability of the software to the support environment, and personnel needed.
2. Revise DoD Directive 5000.29 and 5000.39 to cross-reference each other, explain the relationships between a computer resource plan and an integrated logistics support plan, and require cross-referencing and preparation of the plans simultaneously during the system development process.
3. Appoint an OSD office, such as the Office of the Deputy Under Secretary of Defense (R&AT), to act as a technical advisor to the Defense Acquisition Board on software issues and assist in milestone reviews of computer resource plans.
4. Issue guidance that requires Service-level counterparts of the Defense Acquisition Board to specifically review the computer resource plans, for major and less than major defense systems.
5. Investigate the practicality and effectiveness of establishing individual databases to track software in defense systems. The Databases should be oriented to individual application domains at major system commands. They should include information on development and support costs for the systems the host and target computers in use, the programming languages used, system interface characteristics, system peripheral equipment, software documentation, and software support environments.
6. Require that an annual annex to the Services' Ada implementation plans identify planned new systems and major upgrades included in the Five-Year Defense Plan and state which systems will transition to the Ada programming language.
7. Establish a plan to perform periodic OSD-level reviews of waivers to the use of the Ada programming language, and coordinate with the Assistant Secretary of Defense (Comptroller) on their reviews of waivers to the use of the nine other DoD-approved higher-order programming languages.

E.2.8 Proceedings of the Workshop on Executive Software Issues, Software Engineering Institute, 1988 [SEI88a]

1. Develop a top-level briefing directed at government and industry executives. The briefing should highlight examples of the impact of software on return-on-investment, image, competition, system performance and quality, cost, and schedule. The briefing should include both good and bad aspects of modern software engineering and its organizational effects, as well as the capital investment required to develop a modern software development capability.
2. Inform executives about the flexibility and functionality that software can add to systems.
3. Define the implications of the modular systems development concept, and inform executives of its ramifications.
4. Develop a vision of the nation's direction in software capability.

5. Identify a champion (national level commission, or person) for software to provide leadership in developing and implementing a national strategy for software capability.
6. Define and articulate the complementary roles of industry, government, and academia in technology transition that will result in leading-edge collaboration.
7. Develop a national strategy to attract more people into the software field.
8. Identify the responsibilities of industry, government, and academia in providing appropriate education for the software industry.
9. Develop a briefing and perhaps a videotape for high school seniors, college students, and counselors aimed at career opportunities in the software engineering profession.
10. Create awareness in senior managers of the benefits for retraining and continuing the education of employees for software engineering.
11. Develop a draft program for industry, government, and academia for on-the-job training cooperation.
12. Develop five-year cooperative education curricula in software engineering.
13. Develop example Statement of Work clauses to support acquisition personnel in using new technologies. Example clauses should be developed at a minimum, for the following areas: software quality, reuse, product/process metrics, requirements certification vehicles (prototypes, models), competitive development of requirement specifications, and warranties. The clauses must be developed and tried on programs prior to providing them for general use.
14. Building on already established references, develop a set of common, accepted definitions of software engineering terminology, including terms such as "software reliability," "software quality," and "advanced development models", to be used to guarantee consistency in proposal preparation and in the construction and support of software.
15. Government innovations such as the Software Contractor Assessment Instrument, the Software Engineering Evaluation (SEE), and the Software Assessment should be incorporated into the acquisition process. Corporate executives need to be informed of these assessments.
16. The government oversight role needs to be more clearly described.
17. Better criteria are needed for selecting management indicators. Current indicators often measure what is easy rather than what is important. Contractors should therefore be encouraged to propose and use indicators with which they have had a history of experience and success. Also, the Software Development Plan is an appropriate vehicle for describing the use of management indicators for a particular development or support effort.
18. The Software Development Plan must identify relevant methodologies and tools. It must be a "living" document, updated in a timely manner.
19. The way the government plays its role must be improved. A teamwork relationship between the government and a contractor should be encouraged. Government needs to justify data and review requirements and schedule. Event-driven, as opposed to schedule-driven, reviews are preferred, to assure that reviews of meaningful material are held at an appropriate point in time. The government needs to provide improved guidance to contractors for preparing cited deliverables, so that what is expected is known in advance. Government program office personnel need more software acquisition education and training. The User's Operational Concept (AFR 57-1) is an important input to the requirements process and should be used more effectively.
20. Develop software data rights guidelines to provide a better understanding of data rights as applied to software and documentation within a software reuse library system.
21. Develop policy on the responsibility covering the software and documentation within a software reuse library system. Specifically address the issues of government furnished software as it applies to use in development contracts.
22. Explore policy for a software reuse library system on software warranties.
23. Examine existing acquisition standards (DOD-STD-2167A and DOD-STD-2168) to identify procedures that could create barriers to reuse.

24. Develop a software coding standard for components in the library.
25. Develop metrics for measuring "design for reusability." This metric will enable a potential user to gauge the portability of the components and will provide some quality metrics when comparing similar components.
26. Develop documentation standards for reuse library components.
27. Develop minimal testing standards to be applied to components prior to including them into a software reuse library.
28. Develop standards for documenting the testing (procedures, inputs, results) of components in a software reuse library.
29. Develop the procedures (methods) for analyzing a specific domain to assess areas of high pay-offs in reuse.
30. Develop a cataloging methodology to allow numbering of software and documentation within a software reuse system.
31. Develop a retrieval system for a software reuse library that allows fast, accurate definition of the required component in a user-friendly mode. (The retrieval system may be best approached as an artificial intelligence expert system.)
32. Determine the best approach to the creation of a library system for weapon system software.
33. Determine policy for certification (i.e., software with a trust level of one is highly reliable, level two should be used in all but life-critical applications, etc).
34. Determine policy on allowing access to a software reuse library.
35. Determine policy for handling classified information within the software reuse library.
36. Determine policy for changes to the software in the software reuse library.
37. Determine policy of rewards for programs submitting and using software in the software reuse library.
38. Provide a climate in acquisitions which will promote greater interaction and feedback on software sections of a system.
39. Provide government oversight as a costed and scheduled item.
40. Use more competitive cost reimbursement contracts through the allocated baseline (requirement specification) portion of the effort; this method, which allows more competition, is used in Europe. Use fixed-price contracts once an allocated baseline is determined. This may only work for programs that are well precedented.
41. Identify inhibitors in the current acquisition practices for new technologies.
42. Develop model contract clauses which span the extremes of rights in technical data.
43. To address these problems, evolutionary development methods should be used on all large-scale, complex, critical software systems. These methods should take into account user involvement and total systems implications and should include prototyping and modeling.
44. Change the procurement policies, as necessary, to allow the use of evolutionary development methods for resolution of critical issues.
45. Place greater emphasis and investment on higher quality software products.
46. Executive management should establish, issue, and enforce policies on: (1) a basic set of common, tailorable software processes and environments, (2) software quality and productivity, (3) software measurement, management, and tracking.
47. Software organizations should promptly implement programs to (1) define, collect, store in databases, analyze, and use process data, (2) make quantitative plans and track progress, (3) conduct quantitative risk analyses.
48. Institute an evolutionary development concept through changes, as necessary, to government standards and procurement policies to: (1) encourage the general use of rapid prototyping and consider mandating its use for large, complex software systems, (2) require user involvement throughout the development cycle, (3) require basic software training for users and program office staff, (4) promote a teach atmosphere between program office, developers, and supporters, (5) emphasize the need for systems design continuity throughout software development, (6) recognize the need to maintain a system

design focus throughout the post deployment software support (PDSS) phase.

49. Develop a market for investment in software by providing matching funds for competitive prototyping and software exercises. Use of industrial Modernization Incentive Program money for process and tool improvement should be encouraged. Demonstration of the value of a new technology and identification of requirements for new tools which stem from current and future processes should be funded. Skill and capability being equal, bidders who demonstrate high levels of investment in software development environments, tools, process improvement, and training should be favored.
50. Develop the ability to measure current weaknesses of the software process and the value of proposed improvements. This can be accomplished by: establishing a baseline and periodically updating it; defining and tracking process metrics to understand costs and error sources; and defining and tracking tool-related metrics to understand improvements which lead to significant cost and error reduction.
51. Support selection from alternative software engineering processes and tools. This can be accomplished by instituting a clearinghouse for the evaluation of information which is currently available on industry processes and tools. A tools and processes demonstration and evaluation center for use by the DoD community should be created to support this effort.
52. Require strong error prevention and detection programs throughout the life-cycle. Processes which reduce errors to acceptable levels should be demanded. The development of a program of incentives for early error prevention, detection, and removal should also be instituted. The goal is to achieve warranted software.
53. Change software-related accounting practices by developing a program for limited-rights licensing of support software (including warranties, upgrade plans, etc.), and by developing software depreciation schedules which foster investment in support software. Cost/benefit analyses (including suggested pricing structures) must also be developed. This would allow software development costs to be recovered under manufacturing costs.
54. Provide funding to develop tools and methods to facilitate the use of rapid prototyping.
55. Have organizations such as the Microelectronic and Computing Center (MCC), Software Productivity Consortium (SPC), Aerospace Industries Association (AIA), and the SEI establish a list of software management concepts, system development concepts and technologies, and software methodologies and tools that should be understood by executives. This is required to enable executives to make better software decisions.
56. To improve management awareness of software costs, schedules, and quality, a briefing program should be developed for senior management on benefits of quantitative software management.

E.2.9 Report of the Workshop on Military Software, 1988 [ZRAKET88]

1. OSD should revise and reissue DoD Directive 5000.29, Management of Computer Resources in Major Defense Systems, as soon as possible.
2. OSD should task the Joint Logistics Commanders (JLC) through the Services to provide, as soon as possible, a set of regularly-updated handbooks on the various elements of software acquisition to serve program managers as the implementing guidelines for DoD Standard 2167A.
3. OSD should direct the services to create a software productivity line item that supports the OSD Total Quality Management Initiative for improving manufacturing technology. The capitalization of hardware and software support for software workers should be encouraged and rewarded.
4. OSD and the Services should initiate a joint DoD-industry group to develop changed contractual specifications that allow funding or special award fees for industrial-incurred, front-end software-development risks aimed at agreed product objectives achieved for software reuse, productivity, or quality. Here, the software would be a separate line item in the contract.

5. OSD should issue immediate executive guidance to the Services to supplement DoD Directive 5000.1 to stress a vigorous and substantive effort regarding software during the demonstration and validation phase of every program.
6. Requests for Proposals by the Service product Divisions should require industry to have defined software-development processes that would be evaluated comprehensively as part of the source selective process or generally for more than one competition.
7. Program management procedures for System Program Offices should emphasize the necessary front-end work to reduce risks via:
   a. requirements definition through competitive prototyping and rigorous system design;
   b. incremental implementation - smaller packages, less customizing, and more use of the off-the-shelf programs or components; and
   c. scheduling preliminary design reviews at a time consistent with the degree of validation which has been accomplished.
8. A mandatory, one-day program manager course to showcase aggressive and innovative software-development management techniques such as those outlined above should be developed. This course could be incorporated in the Defense Systems Management College (DSMC) and/or given as a Technical Management Refresher Course by the Services and helped by the use of video tapes to augment live instruction.
9. OSD should direct the DAR Council to iterate the current draft of the Revised DAR Rights-in-Data Clause (based on the recommendations of the recent SEI Report on this subject) and to issue it as soon as possible this year.
10. DoD validation centers should be established for certain classes of software similar to the ones for multilevel secure operating systems (at the NSA Computer Security Center) and Ada compilers (at the Ada program office). However, the necessary technical foundation for these centers must first be established.
11. OSD, with DARPA and the Services, should explore the technical issues involved in validating significant properties of the classes of reusable software components. OSD should then see to the establishment of the technology base necessary for cost-effective validation and the evaluation activities needed for such classes of reusable components. The architecture of standards needed to define component interface specifications should be included in this effort.
12. Current near-term and long-term DoD technology efforts in the area of software engineering environments are today's keys to DoD's software productivity and quality objectives, and should be continued.
13. A unified DoD Software Technology Plan should be established by OSD including funding needs and plans showing how DoD's near-term and long-term technology efforts fit into a unified strategy for increasing software productivity and quality. Where investments are not being made in technologies that may improve software development, additional investments should be made.

E.2.10 Army Materiel Command Study, 1989 [AMC89]

1. Implement an Effective Software R&D Strategy: The Army must recognize that, because of the growth of the commercial software market, it needs to be primarily a buyer rather than a builder of technology for software development. The Army strategy for software technology should be built upon the industry standards and use of commercial tools whenever possible. However, in those areas where special Army needs exist, the Army needs to pursue an aggressive, focused R&D program.
   — Develop a Software Technology Plan
   — Establish a testbed to evaluate commercial products
   — Evaluate commercial products/standards for Army applicability
   — Structure incentives to increase compatible Industry Software IR&D
   — Conduct research to satisfy specific unfulfilled Army needs
   — Structure R&D program with reuse as a keystone

2. Develop an Approach to Software Reuse: A consistent approach to software reuse must be developed. Implementing effective software reuse procedures will result in cost savings, improved quality, and reduced development time. The approach must be built on the recognition that it will take at least ten years for reuse technology to mature. Initial efforts will employ reusable software artifacts, follow-on efforts will improve ways to adapt existing systems. In the final phase, reuse will be a function of the tools used to generate new systems.
   — Develop a plan to acquire reusable software
   — Establish a software reuse R&D program
   — Investigate non-technical issues
3. Develop and Evaluate Software Metrics: Process and product oriented metrics need to be defined and evaluated. Tools which support useful metrics should be integrated into software environments used to develop and support Army software. Quantitative measures of contractor performance and product suitability are needed to ensure successful management of the software process.
   — Collect and use existing compiler performance metrics
   — Establish framework to evaluate metrics for applications
   — Calibrate existing metrics with ongoing programs
   — Develop and evaluate new product/process metrics
4. Evaluate Software Life Cycle Models: Software technology is an immature, rapidly evolving technical field. Dramatic growth in complexity and size of Army software systems requires the Army to foster and direct the evolution of new practices, procedures, and methods for the development of software systems.
   — Develop model to select paradigm and evaluate payoffs
   — Investigate alternative paradigms
   — Improve specification correctness/completeness analysis methods
   — Investigate other supporting technologies
5. Establish Controls on Software Environments: The Army must develop a viable approach to the management of software engineering environments it uses or permits to be used for MCCR development and support. Initiative and productivity of developing contractors needs to be encouraged, yet the Army must ensure that systems are supportable. In-house software support environments need to be standardized to achieve economies of scale, improve resource efficiencies, allow more rapid transition to a support posture, and improve productivity and quality.
   — Establish requirements and standards for developers
   — Develop requirements for extensible Army support environment
   — Constrain target machines to meet Army needs, reduce cost
6. Manage the Introduction of Ada into the Army: Ada introduction plans and activities need to be strengthened if the efficiencies and economies of Ada are to be achieved. The use of Ada will reduce the number of tools required in the Army's support environment, improve productivity, and increase quality of software produced. No Army strategy for the control of Ada and its introduction has been evident. An effective and purposeful approach is needed.
   — Fund an Army supplement to the OSD Ada Technology Insertion Program
   — Evaluate efficiency and utility of commercially available tools
   — Develop complete Ada training program within Army
   — Evaluate success of Ada Insertion
7. Establish Mechanism for Reverse Engineering: Standards for computer resources including software languages, hardware design, documentation, and configuration control have evolved since its first application to weapon systems. In addition, many existing systems were developed in a schedule driven, resource constrained environment. Because of these factors, the Army must recognize the need to use reverse engineering to understand system design from existing software and documentation.

— Address need for reverse engineering in planning support
— Establish criteria for level of reverse engineering
— Investigate use of evolving technology to assist in reverse engineering

8. Develop a Strategy for Technology Insertion: The Army must improve its software state-of-the-art practice to meet the needs of the large and complex mission critical computer systems of the future. These improvements must be promulgated within the legal, fiscal, and contractual constraints of the government and reduce the risk to system development accruing from the use of unproven technologies.
— Identify specific risk management funds for software
— Fund parallel developments when introducing new technology
— Provide contract award for successful technology application
— Establish transition points and mechanisms
— Develop techniques for Software Process Improvement

9. Conduct Integrated Software Planning: Computer Resources Management Plans (CRMP) do not serve their intended function as currently prepared because they do not address critical issues and are not integrated into the system acquisition planning process. Planning for software must be addressed from the total life cycle viewpoint, with proper attention being given not only to initial development, but also to the critical aspects of software maintenance and improvement.
— Streamline Computer Resource Management Plan
— Computer Resource Working Group provides forum for PM
— CRMP documents conditions of eventual software support
— Approval of CRMP by PEO/AAE ratifies strategy

10. Tailor Software Acquisition Process to Systems: The Army needs to encourage the use of alternative software development models rather than the rote application of existing standards. The vast differences in the software that the Army buys as well as the limits of the "waterfall model" must be recognized and deliberate steps taken to reduce acquisition risk. Procedures are needed to: refine requirements prior to design, strengthen the design process, emphasize "software first," clarify design parameters, and improve the user/developer interface.
— Establish multi-axis acquisition classification scheme
— Define candidate strategies for different system classifications
— PMs classify systems and use classification to structure acquisition
— Ensure risk areas addressed before Full Scale Development

11. Develop a Consistent Contracting Approach: All too often, software received late consideration in the contracting process. The time to establish specific requirements, get contractor commitments, and ensure adequate resource allocations is prior to contract award. Procedures need to be established to reward competent contractors, force an early consideration of development plans, and negotiate effectively for software consideration during development.
— Address software in proposal preparation instructions
— Evaluate contractor software maturity in source selection
— Incorporate software performance as part of MACOM database

12. Organize Army to Manage Acquisition Process: Make realignment on Army staff to provide effective Army Acquisition Executive control over the acquisition of Army systems, especially those which rely on mission critical computer resources. Clear management control is needed to: improve management practices, unify Army staff into an efficient structure, and develop a credible advocate for computer resources to Congress and the national leadership.
— Eliminate bicameral MCCR management at Headquarters, Department of the Army
— Correct AR 70-1 so it applies to information handling systems

13. Improve PM/PEO Computer Resource Management: Establish an effective working relationship with closer cooperation between the PM/PEO and their supporting MACOMs. The present system, which has given PEOs independence from MACOM

policy and guidance, must be changed if weapon system software management is to be improved.

- Dual hat functional commanders as Program Executive Officers
- Create CRWG early to identify problems in CRMP building
- Use CRMP as sole basis for computer resource strategy decisions
- AAE establish process to stop systems with ill-conceived CRMP
- Hold LCSE directors responsible for raising planning deficiencies
- Use "contracting authority" as required
- Provide experts to advise PEO; report to MACOM/AAE

14. Establish Clear Organizational Responsibilities: Provide for a clear understanding of organizational responsibility at all levels within the Army. Changes are needed to: delineate the roles of the major organizational elements in the area of computer resources, implement a cost effective and cohesive organizational structure, provide clearer lines of management within the Army, and prevent duplication of effort in the various organizations.
    - Define role of HQDA in overall management
    - MACOMs enforce policy and define strategy
    - Major Subordinate Commands support acquisition
15. Strengthen AMC's Software Management Role: The AMC organization needs to take into account the importance of mission critical computer resources to the Army. The command must manage its computer intensive systems so they are reliable, meet user requirements, and are supportable during their life cycle. In order to do so it should be resourced to manage the increasing role of computer resources in weapon systems, provide a strong advocate on the AMC staff, and provide career paths for software professionals.
    - Establish well resourced, limited life Special Operations Center
    - Create an Assistant Deputy Chief of Staff for MCCR
    - Establish senior level MCCR S&T advisor to Commander
16. Provide One-Stop Support for Project Managers: The scarcity of computer hardware and software experts within the Army makes it critical that the available people are used effectively, provisions are made to nurture and develop a competent staff, and functional duplication is eliminated. The Life Cycle Software Engineering Centers should become the responsible activity to ensure that this happens. As such, they must become the single source of software support for PMs.
    - LCSE responsible for in-house software engineering
    - LCSE provides services to PMs, not a "body shop"
    - PM/PEO staffs limited to managers not doers
    - PMs/LCSE ensure software visibility during development
    - LCSE maintenance activities under rigorous controls
17. Build an Army Software Technology Center: The trend to disperse the critical mass of technologists supporting software and to decrease the annual research and development budget for software technology must be reversed. The Army needs an integrated, effective approach to software technology which will provide a critical mass for software tools and technology, serve as a vehicle for technology insertion, and insure responsiveness to Army wide MCCR needs.
    - Reaffirm decision to have STC at Ft. Monmouth
    - Establish critical mass of people and funding for 5 years
    - Establish resource source, concentrate other activities
    - Create a software technology affiliates program
    - Run Software Engineering Intern program as part of STC
    - Define specific technology insertion tasks and controls
    - Assign technology proponency to STC; advocacy at HQ
    - Develop technology program with maximum use of commercial base
18. Organize to Grow Software Engineers: Recognize that software engineers have different skills and abilities than others. Army must plan to grow its own Software Engineers from

within and also needs to ensure their effective use. Provide a mechanism to provide both technical and domain maturity before putting Software Engineers into management positions.

— Use new GS-0854 series; do not permit grandfathering
— Provide four distinct levels of performance
— Establish opportunities for Senior Software Engineers
— Use co-op program to Identify Outstanding Candidates

19. Eliminate Confusion in Training Device Support: Because transition of life cycle support for training devices and systems has been difficult to achieve, AMC must ensure that an organizational structure is in place to provide the life cycle software engineering support. A solution needs to take into account the problems of resourcing, support system specific devices, interoperability, and the inherent difficulty in building a software support capability.
— Assign life cycle software engineering responsibility
— Guide process by following rules
— Test use of Total Contractor Software Support as alternative

20. Provide Virtual Colocation with TRADOC Centers: Where AMC and TRADOC centers are colocated, the communication between the two is generally excellent. Where the centers are physically separated, communication suffers. Communication in a wide variety of areas must be improved. Areas of importance include: problem identification and tracking, requirements understanding, configuration management, and test participation.
— Use electronic means to provide virtual colocation

21. Develop Pilot Software Awareness Program: The Army needs to publicize: (1) real and near real-time software's pivotal role in fulfilling Airland Battle Doctrine, (2) software engineering role in developing and maintaining efficient, effective, and economical combat software. Awareness of software's force multiplication aspects will support resource allocations at the highest level of government.
— Develop Airland Battle Software Story
— Prepare an Airland Battle Software awareness brief
— Brief key Defense leaders on software role/initiatives
— Solicit and record feedback information

22. Develop Operational Software Literacy Program: Army needs to develop an Airland Battle Software Awareness/Literacy program for congressional, OMB, OSD, Joint, and Army leadership which will: (1) elevate their consciousness level with respect to software's pivotal role in winning the Air-Land battle in the 1990 and beyond timeframe, (2) address software awareness/literacy within the Officer and Non-Commissioned Officer Corps, and (3) harness the creative potential of enlisted personnel in using software as a force multiplier.
— Build Software Literacy program based on Pilot Awareness Effort
— Execute Literacy Program
— Get to General Officers to show impact of software

23. Find Army Software Advocates: Computer technology budget has decreased by order of magnitude in last five years. An advocate is needed at both the MACOM and ARSTAF levels. Additionally, proponency for Life Cycle Software Engineering appears confused with weapons system support having no effective proponent.
— Find fighters for software technology at AMC & Department of the Army
— Correct failure of AMC proponent to support MCCR

24. Establish clear Acquisition Policy for Software: The Army should provide a clear, unambiguous implementation of DoD Directive 5000.29 for Mission Critical Defense Systems. Chapter 8 of AR 70-1 establishes the basis for such a policy, but it needs to be implemented and remaining ambiguities with the AR 25-series regulations needs to be removed. Realistic policies and controls applicable to PEOs and PMs need to be implemented.

— Establish clear and concise definition/process for MCCR

— Create integrated policy stream under AR70-1 for MCCR

— Integrate computer resource issues into PM/PEO/AAE process

— Require all approvals and waivers in single document

— MACOMs maintain database on computer resource requirements

— Provide implementation in AR 70-series regulation

— Require consideration of life cycle tailoring

— Provide guidance for evolving new paradigms/environment strategy

25. Clarify Funding Policy for Software Support: Need to obtain clear-cut and unambiguous guidance on LCSE funding policy that will provide the most efficient management of LCSE functions. Recent funding policy changes have streamlined the process, but several residual issues must still be addressed.

— Determine where to report manpower needs in BPRR/MAMP

— Consider augmentation of OMA core funding in MDEP MS2B

— Limit use of MCM for software improvements

26. Provide for Management of Software Change: Individual software changes to systems need to be identified, costed, prioritized, and approved through a disciplined change approval process. Although costs for systems will be estimated based on the best available models, the OMA P7M funds which are identified need to be expended to get the best possible value to the Army. A joint prioritization must serve as the basis for allocation of funds and identification of deferred software maintenance and improvement tasks.

— Classify software support tasks into two simple categories; maintenance and improvements

— Define a minimum level of sustainment for each system

— Conduct an annual joint AMC/ISC/TRADOC prioritization

— Complete review early so that funds can be reprogrammed

27. Establish Internal Controls and Feedback: Internal controls within MACOMs need to be used to minimize the risk of having software material weaknesses. In general, existing controls have failed to provide feedback and corrective actions. Actions have been primarily driven by outside audits, studies, and reports. Each MACOM should establish a management and control process to identify and correct systemic weaknesses regarding the development and support of mission critical software.

— Establish a formal process to identify/investigate problems

— Create database to track software issues/problems/solutions

— Assign responsibility and demand accountability

28. Develop a Computer Resource Data Base: Today's major problems with software development are not basic technology problems, but failures in management. A major re-examination and change in attitudes and practices concerning software acquisition is needed. A key part of that change in attitude must be a more comprehensive view and assessment of the computer resources used in MCCR systems.

— Develop a database for system computer resource information

— Establish capability to feed Major Subordinate Commands' databases via DDN

29. Enhance Interaction between Activities: Periodic and ongoing activities should be used to improve communication and foster interchange of information between Army and other DoD activities with an interest in mission critical software. The Life Cycle Software Engineering Steering Committee (LCSEC) should be revived to foster cooperation between Army activities. Support to the Joint Logistics Commanders' Committees should be expanded to best utilize the cooperation between the services on policy, technology, planning, and software support matters.

— Re-establish quarterly meetings of Army LCSEC Commanders

— Provide regulars General Officer meetings with Steering Committee

— Expand support for JLC Software panel

30. Integrate Software Quality into Process: American quality organizations are typically considered "second class" operations. By focusing on engineering the quality into the

design rather than the "assurance" aspects, the Army needs to force quality into a position of preeminence. We need to ensure the credibility of our quality organizations by using fairly senior people with solid software credentials.

- Fully integrate quality into software development process
- Hold LCSEC responsible for managing software development
- Require LCSEC to establish internal quality controls
- Hold PA&T (Program Analysis and Test) responsible for process control

31. Improve Software Configuration Management: Configuration control of software and management of those configurations has been based on existing hardware regulations as implemented by the various subordinate commands. No standard configuration accounting systems or even software numbering systems have been selected. Standardization activities need to be pursued.
    - Work toward a standard configuration management tool
    - Institute a standard Computer Program Identification Number
    - Implement standard tiered interoperability control board
    - Facilitate Software Reuse
32. Implement Effective Interoperability Control: Army needs to enforce a top-down approach to develop, plan, and refine baselines to support a "system of systems" approach to interoperability. Concepts which are now stated at a high level must be refined to define, model, evaluate, and control system to system interfaces. Interoperability evaluations cannot be deferred until operational tests. A hardware basis is needed for component integration below the level of the command and control nodes.
    - Create Army Interoperability Executable Model
    - Accelerate funding of Army Interoperability Network
    - Build Government interoperability knowledge base
    - Investigate component integration/standardization
33. Address Software as part of Materiel Release: Current regulations permit, but do not specifically require use of the Materiel Release process for software. The resulting confusion needs to be resolved with a clear statement that the release process be used to release all block improvements to software.
    - Revise AR 700-142 and AMCR 700-34 to address software
    - Evaluate and recommend procedures for special/evolving needs
34. Develop Responsive Distribution Process: AMC tactical computer systems increased from 85 systems in 1980 to 232 systems in 1989, and will continue to grow. The standard logistics supply system is not adequate for supporting software change distribution, especially when major modifications to interoperability command and control systems must be accomplished. Current method of sending teams from the LCSEC will be impractical as the number of systems continues to grow. Alternative methods need to be investigated now.
    - Conduct experiment with forward replication/distribution
    - Develop Army go-to-war strategy for software upgrades
    - Require consideration of externally programmable memory
    - Develop regulation addressing software distribution
35. Provide Software Maturity Management: Systems must be managed so we avoid a "final exam mentality." The Army does not now have, but needs to use an approach for tracking the maturity of software in systems. Deficiencies must be identified and corrective actions taken before the system reaches its formal testing phase. It is critical that the focus on system testing be lessened. The Army must ensure that component tests are properly structured and that the information from them is used to identify and remove deficiencies.
    - Develop, evaluate, and then use software metrics
    - Require use of approved monitoring tools
    - Use system approach to show intermediate results
    - Reach consensus for on-going evaluations

36. Enforce Standard Software Cost Model Use: A variant of Boehm's COCOMO software cost estimating model has been developed for Army use in Life Cycle Software Cost forecasting. The model, called SECOMO, was validated but different versions are starting to appear. A standard, approved version should be maintained. In addition, further refinements to the model need to be addressed and a methodology for forecasting development, in addition to support costs, needs to be developed.
    — Mandate use of one authorized version of SECOMO model
    — Support further refinements to the model
    — Determine areas where further improvements are necessary
    — Conduct research to develop model for use in development
37. Improve Interface into PPBS for Software: Establish a capability to capture total LCSE requirements and latest funding guidance from multiple commands and appropriations. Isolate and track LCSE costs through the PPBS process.
    — Design a process to capture LCSE requirements/guidance
    — Provide timely feedback from PPBS decisions
38. Identify and Capture Actual Software Costs: In spite of the ever increasing cost of software to the Army, it is not possible to identify and track those costs. Actions need to be taken to collect software costs both during development and during the support phase of the life cycle. It must be recognized that collecting hardware and software costs together does not provide sufficient visibility into the development process.
    — Develop common data definitions across services
    — Require contractors to isolate and report software costs
    — Establish standard procedures to report in-house software cost
    — Develop policies and instructions concerning cost identification
39. Provide Efficient Front End Loading: The Army evolved the concept of Post Deployment Software Support into Life Cycle Software Engineering approximately five years ago. This action provided additional consideration of software engineering at the front end of development rather than waiting until it was time for support. With more resources required to support the increasing number of transitioned systems, it is time to refocus resource allocation to emphasize early, high leverage actions.
    — Define specific front end tasks to be done in-house
    — Determine methods to resource front end tasks
    — Establish Army sponsored Federally-Funded Research and Development Center (FFRDC) for acquisition assistance
40. Consider Alternative Support Options: The concept of in-house Army software support envisioned over ten years ago was never executed because of resource constraints. Typically, each LCSEC uses a support contractor to perform maintenance and improvements on the systems it manages. Sometimes government facilities are used, but other times they are not. There is a need to consider alternative support concepts with the purpose of minimizing cost and freeing up government people to focus on emerging systems.
    — Pick selected systems to test alternative concepts
    — Assess cost and risk of promising alternative concepts
    — Establish guidelines to determine optimum concepts
41. Conduct Contracting Out Study: Army still does significant in-house development and support of ADP/MIS systems at Central Design Activities. These systems are much more similar to commercially available systems than those embedded in weapon systems, and the Army may be mis-allocating its people by focusing its talent on these areas while giving short shrift to its tactical systems. A complete evaluation of the feasibility of contracting out these ADP/MIS activities should be conducted.
    — Evaluate feasibility of freeing up TDA positions for MCCR
42. Measure Efficiency of Current LCSE Centers: The decision to use a controlled number of LCSE centers is based on a study which is over ten years old. No data is available to help evaluate the effectiveness and efficiency of these centers. Productivity data should be

collected and used to update PDSS concept study.
— Identify efficiency measures for LCSECs
— Institute on-going data collection effort

43. Develop Software Engineering Career Program: Provide a career program with direct and tangible benefits to employees. There must be convincing evidence encouraging them to enter and stay in the field. Such a program will include the following features: strict standards to enter and progress in the program, effective career management, formal and continuing process of training and development, and good opportunities for high-level career progression.
— Use new GS-0854 series for software engineering
— Establish precise qualification and certification standards
— Establish target jobs at various professional levels
— Implement dual track system with equal rewards
— Provide tangible incentives at each level
— Get salary levels competitive with industry

44. Improve Incentives for Military Software Experts: The Army needs to recognize the importance of Software to its war fighting capability and stop discouraging and frustrating those young officers with software talent and education. A process needs to be developed to ensure software capability is used as a criteria when assigning Program Managers to computer intensive projects.
— Broaden career path to General Officer
— Provide software understanding for 515
— Enhance 53A classification; use 25B instead
— Provide for functional automators
— Treat software intensive positions as command assignments
— Advanced Education Requirements Board identify masters degree in software for MCCR PM positions
— Provide additional software intensive add-on to DSMC PM course
— Accredit the U.S. Military Accademy Software Engineering Department

45. Establish Career Subprogram Management: A strong career management infrastructure is necessary in order for the Army to attain maximum return on investment in Software Engineering personnel. As the job series for Computer Engineers is implemented, intensive management will be necessary to ensure that proper and effective standards are developed, only well qualified engineers are admitted to the program, each software engineer's technical and managerial maturation is planned and executed.
— Establish Software Engineering Subprogram manager
— Establish network of Software Engineering Mentors
— Establish temporary Career Management Staff
— Write E&S ACTEDS Master Training Plan

46. Provide Job Challenge for Software Engineers: Successful complex software programs use a government acquisition force 10% of the size of the contractor's software development group. Army systems seldom can muster a force this large. Need to effectively use the people we have, yet realize that they need some hands-on experience to maximize their competence.
— Channel Software Engineers into high leverage activities
— Develop means to maintain proficiency
— Identify specific skills and assess as part of IDP reviews

E.2.11 Software Technology Development and Deployment Plan for the DoD Technology Base, 1989 [IDA89]

1. Establish a comprehensive approach for improving military software during deployment. The focus should be on the phases of the software life cycle that have the most influence over the deployment of software intensive systems. The approach should include a series of demonstration projects that focus on developing system upgrades for operational

software intensive weapon systems currently experiencing critical software difficulties.

2. The Service Laboratories' software technology base programs and the DoD Software Initiative (Ada, Software Technology for Adaptable, Reliable Systems (STARS), and Software Engineering Institute (SEI)) should continue to be supported and fully funded for FY 1990. Since the rate of software technology evolution has not kept pace with DoD weapon systems' growing requirements for more software with increasing complexity, DoD should reassess the software technology base requirements for FY 1991 to ensure that the software technology base funding priority is increasing commensurate with the trend of increasing software-related deployment costs.
3. The Services and Agencies should be tasked to establish "lead agents" for selected critical software technology areas. The lead agents should be carefully selected from within the Services for their expertise in software-related disciplines. The primary functions of the softwar lead agents would be to identify Service software technology related problems and requirements, to share software technology information with DoD and the Services and to stimulate advanced software technology application across Services and program offices.
4. Establish an immediate initiative to consolidate DoD software and computer policy, management, and oversight within a structured forum (with both responsibility and authority) that is commensurate with software technology's growing importance. The following were recommendations developed during the workshop to specifically address this very complex software-related responsibility and authority problem.
    a. Establish a Directory of Military Software Improvement, within the Office of the Under Secretary of Defense (Acquisition), responsible for identifying, managing, integrating and implementing software deployment policy.
    b. Establish a committee under the co-chairmanship of the Director Defense Research and Engineering and the Assistant Secretary of Defense (Production and Logistics) with representation from the Assistant Secretaries of the Services. The committee will provide a direct forum for addressing policy and program issues that transcend Service and Agency requirements.
    c. Develop and distribute a software responsibility and authority guide. The intent of the guide should be to clearly outline the various software-related roles and responsibilities within OSD and permit a greater understanding of the current software policy, review, coordination and oversight related processes within DoD.

E.2.12 Adapting Software Development Policies to Modern Technology, Air Force Studies Board, 1989 [AFSB89]

1. AFSC, working with the JLC organization, should ensure that development models and accompanying rationale alternatives to the waterfall model, based on risk reduction concepts, are included in the forthcoming Handbook 287 for DOD-STD-2167A, with supporting direction in AFR 800-2 and 800-14.
2. AFSC should ensure adequate software risk reduction for unprecedented systems during a Demonstration/Validation or equivalent phase preceding full-scale development. For unprecedented systems, AFSC should provide policy guidance for competitive two-phase procurements, such that software risks are reduced to a practical minimum before proposals are prepared for the following phase(s). For unprecedented systems, AFSC should direct an independent assessment of system and software risks near the end of the Demonstration/Validation phase or Phase 1 of a two phase procurement, as part of the basis for follow-on procurement decisions. For multiphased procurements, AFSC should direct that contract mechanisms provide continuity of essential contractor skill and expertise between contract phases.
3. For key technologies in systems and application areas where operational threats or requirements change rapidly, AFSC should fund parallel technology programs at the system level to foster a ready industrial base from which to compete single phase system acquisitions.

4. Each program involving software should be required to carry out early identification of critical software issues, and to develop and maintain a Software Risk Management Plan. This instruction shouid be included in the most appropriate Air Force management directives.
5. When a program manager is faced with late identification of software requirements that can be deferred to a later time or capability block, AFSC management guidance should encourage and support this deferral and accept the consequences of doing so.
6. User involvement should be tailored for each program, varying from cases requiring very limited involvement to ones in which a user will assume the lead role.
7. AFSC should direct its product divisions to tailor the contract form for each specific program's needs; in particular, AFSC should avoid using firm fixed price contracts for unprecedented programs. (This will require management follow-up, consistency, and the support of higher authorities.)
8. Product divisions should be directed to specify use of an SEE for each program having, as an example, a software staff of more than 12 people, and to require proof of its existence and the contractor's knowledge of its effective use, in order to qualify.
9. When software development contracts are granted to design groups that are organizationally or geographically separated, near-term management criteria in source selection should emphasize use of modern telecommunications and division of tasks to reduce requirements for interface among separate locations or organizations. In the long term, contractors should demonstrate availability and use of a software management and engineering environment designed for such dispersed efforts.
10. Product divisions or HQ AFSC should regularly monitor CRWG performance. Explicit evaluations should be solicited from using commands and AFLC.
11. AFSC, with AFLC and the using commands, should sponsor a fresh look at actual maintainer documentation needs. This review should consider the growing automation of documentation by contractors, and how that might be used to reduce the cost or improve the utility of the data.
12. AFSC should select an appropriate program (or programs) through which to implement incremental acquisition, using it (or them) to articulate to OMB and the Congress the need for and special benefits of an evolutionary, incremental acquisition approach.
13. AFSC must strongly encourage AFLC and the using commands toward collocated support for software in integrated systems, rather than complex reprogramming without adequate resources in the field.
14. AFSC should broaden the base of its personnel skilled in acquisition of software-intensive systems; prepare, use, and maintain current guidebooks; and exercise special management of the skilled personnel resource.
15. AFSC special management of software skills should include a software systems engineering advisory team (a "nest of owls"), and special career tailoring for selected officers and civilians.
16. AFSC should take steps to increase the motivation for innovative acquisition tailoring. AFSC should issue a policy statement, conduct workshops, and distribute guidebooks.
17. AFSC, in collaboration with others, should make available to officers and civilians a mid-career systems engineering and software engineering graduate program and appropriate short courses.
18. The Air Force should consider revision of AFR 800-14, paragraph 5-3, Test Planning, and all derivative directives, to require demonstration of testing of every instruction within the software prior to completion of development, test, and evaluation (DT&E). Implementation needs, cost, and expected benefits should be analyzed by experts prior to implementing this revision.
19. Each program should consider using the designated software "maintainer" (operational phase) as the IV&V agent during software development.
20. AFSC, with the Joint Logistics Commanders, should expedite preparation and distribution of the 2168 guidebook, and support maintenance of this and other software guidebooks

over time.

21. AFSC should require the use of commercial off-the-shelf software test technology in system and software development, and make it a part of the technology and software process R&D programs to further advance the area, and apply it throughout the software life cycle.
22. AFSC should select key programs that have high concerns for reliability, maintainability, reusability, interoperability for demonstration and evaluation of this prototype product quality assessment scheme. AFSC should invest funds to merge product and process quality measurement schemes to get increased benefits, and to keep the measurement technology updated to the needs of future life cycle models.
23. AFSC should initiate a program in the style of MANTECH to transfer software development process technologies into actual major system and software development programs.
24. AFSC should increase its technology base investment in software engineering technology, which is currently running at less than $8 million per year. This increase should involve Air Force laboratories more broadly and directly than in the past, and include AFSC-wide coordination. The rate of increase of the technology program should be significant, but must be based on the capacity of the technology resources in defense industry and research institutions with coupling to systems developers to grow gracefully. As a way to improve software technology transfer, and in line with its usual strategy, AFSC should select system programs for application and demonstration of advances in software engineering technology, and provide separate 6.3 funding to support demonstrations.
25. The AFSC should consider funding a program to evaluate candidate SEEs and where applicable, stand-alone tools, for consideration as acceptable environments and tool sets.
26. AFSC should create and fund a project to provide support for the software systems engineering advisory team(s) of Recommendation 8, in particular to capture the knowledge gained and used by the team members for use via knowledge-based tools. This could be a valuable lead project for later use of similar tools, more broadly in AFSC system and software acquisition management.

E.3 Actions in Response to Recommendations

The actions listed here include all types of responses to recommendations: directives, regulations, reorganizations, new initiatives, etc. The sources for information about these actions were communications from members of the Working Group. The actions are generally categorized by organization, e.g., Army actions, OSD actions, etc. Some of the actions included here describe already existing conditions. Thus, if a recommendation was made that is already covered by an existing directive or regulation, that is so indicated.

E.3.1 OSD Actions

1. STARS and SEI are now together under DARPA
2. The Defense Acquisitions Board (DAB) Science & Technology Computer System Working Group has been formed to define a Software Master Plan.
3. DoD published DoD Directive 3405.1, "Computer Programming Language Policy" April 2, 1987, to replace DoD Directive 5000.31 and it includes Ada as one of the approved DoD higher order languages. It also published DoD Directive 3405.2, "Use of Ada in Weapon Systems," March 30, 1987, which mandates the use of Ada in **all** software applications, except previously developed software, commercial off-the-shelf (COTS) software, and as a test language for automated test of hardware components.
4. Metrics and measuring techniques for the software development process and software product quality are required to be identified in the Test and Evaluation Master Plan (TEMP) (DoD 5000.3-M-3) and are an integral part of the T&E process (DoD Directive 5000.3)
5. Initial metrics for implementation progress have been developed (AFSCP 800-43, DoD 5000.3-M-3).

6. DoD 5000.3, 5000.3-M-3, and 2167A require program management procedures for System Program Offices to emphasize front-end work to reduce risk.
7. Baselines, designs, reviews, tests, and requirements are addresses in MIL-STD 483A and 1521B.
8. DoD Directive 5000.31 has been replaced by 3405.1 & 3405.2
9. DoD Directive 5000.29 is currently under revision
10. DOD-STD-2167A has been issued
11. DoD-HBK 287 has been reoriented to tailoring guidance for DOD-STD-2167A
12. A Blue Ribbon Panel is being established to study DoD corporate information management.
13. The AJPO funded a study of cost models; this study showed that a single standard model was probably not feasible.
14. The latest release of the Ada Compiler Validation Capability (1.11) includes mandatory tests for certain Chapter 13 features.
15. The STARS Program was initiated in November, 1984.
16. The 2168 Guidebook has been published as MIL-HDBK-286.
17. The SEI has published several reports concerning the evaluation of software environments including *Evaluation of Ada Environments* [Weiderman87], and *A Guide to the Classification and Assessment of Software Engineering Tools* [Firth87].
18. An executive level group is being established to review and recommend changes to DoD policy and procedures applicable to the development and management of defense information systems and software (DEPSECDEF Memorandum, "DoD Corporate Information Management," 4 October 1989). This group is tasked with examining current DoD systems for potential commonalities within functional areas with an objective of reducing the number of service agency-unique applications. It is also reviewing the adequacy and appropriateness of current DoD policies to today's defense environment, and is reviewing current DoD policies related to the management and development of information systems and software.
19. The Ada Technology Insertion Program (ATIP) was established to encourage the use of Ada in the modification and development of systems and to mitigate potential barriers to the adoption of Ada for systems development within the DoD.
20. The Defense Acquisition Board (DAB) has approved incremental development strategies for both the WWMMCCS ADP Modernization Program and the Defense Message System. Both programs have adopted evolutionary development methodologies which emphasize the use of prototypes and mandates the use of Ada and OSI protocols, as well as other standards, to reduce technological and life-cycle risks.
21. Real-time performance and other Ada performance issues have been addressed in the Ada Compiler Evaluation Capability (ACEC) test suite available to the DoD community. Other similar performance tests developed in the defense industrial sector are available to the government which provide similar information regarding the evaluation of Ada compilation systems (e.g., Hughes Benchmark).
22. The "Ada 9X" effort is designed to revise the standard based on recommendations from the Ada community; the revision will be in accordance with DoD and ANSI procedures.
23. Development of a Common Ada Programming Support Environment (APSE) Interface Set (CAIS) has been accomplished.
24. The Defense Data Network (DDN) is required for use by all ADP systems and data networks requiring data communications services. The objective of the DDN mandate is to provide communications services to the defense community which are reliable, secure, and interoperable. DoD has also mandated the use of GOSIP protocols in order to improve interoperability among applications systems and migrate the DoD community towards the concept of open systems architecture.
25. DoD Directive 5000.37 addresses the issue of COTS, but should be revised to account for changes since its 1987 release. The mandate to use GOSIP through its adoption of commercially and internationally accepted standards, accomplishes the use of COTS

products, and increases the likelihood that COTS networking products will be used to meet DoD networking and communications needs.

26. There are several sources of reusable (Ada) components available to DoD users: Common Ada Missile Packages (CAMP) components developed by McDonnell Douglas, Generic Reusable Ada Components for Engineering, Software Components with Ada (Booch Components), as well as Ada components available through the Ada Repository.
27. The DoD Computer Institute run by the National Defense University is developing a five-day course specifically tailored to program managers of major automated information systems. The course will review management oversight responsibilities for these systems.
28. Numerous security capabilities, evaluated and approved by the NSA, are commercially available. The NSA's Evaluated Products List identifies manufacturers of products which are designed to provide security capabilities for software applications, operation systems, networks, etc.

E.3.2 Army Actions

1. "CECOM Software Acquisition and Support Policy for Mission Critical Defense Systems (MCDS)", was issued 17 July 1989. The purpose of this memorandum is to establish Communications-Electronics Command (CECOM) Software policy for MCDS. The motivation behind this policy is to help minimize risk and cost in acquired software and to promote greater commonality across systems.
2. "CECOM Regulation 11-31, Computer Resource Management Policy", was issued 11 October 1988. This regulation establishes the CECOM Center for Software Engineering (CSE) as the Computer Resource Manager for all MCDSs managed or supported by the CECOM.
3. "CECOM Regulation 11-25, Ada Policy," was issued 1 June 1988. This regulation establishes the requirement for the use of the Ada programming language for all U.S. Army CECOM developed, managed, or supported MCDSs and implements DoD Directive 3405.1 and DoD Directive 3405.2 within CECOM.
4. "CECOM Regulation 700-55, Replication and Distribution of Computer Program Sets", was issued 30 May 1989. This regulation establishes policies and identifies responsibilities for the management of the replication and distribution of computer software and firmware for Battlefield Automated Systems supported by the U.S. Army CECOM.
5. The CECOM CSE is preparing a Master Plan, currently in draft form. The plan encompasses (both external and internal) Life Cycle Software Support operations performed by the CSE. The plan consists of a Core document supported by a number of Annexes covering subjects such as: Management Information Systems, Replication and Distribution, Installation, and Training; Procurement; Strategic Long Range Plan; Total Quality Management; Quality Assurance; and Configuration Management.
6. The Commander CECOM has issued a policy (29 July 1989) providing Ada compilers to academic institutions which will motivate establishing Ada curriculum thereby providing an increased work force knowledgeable in Ada.
7. CECOM CSE has initiated an effort in process metrics to provide managers greater visibility into the software development process.
8. AMC-Regulation 70-16A (Draft) was issued 16 August 1989 which, when approved, supersedes DARCOM Regulation 70-16 dated 16 July 1978. This regulation establishes policy and assigns responsibilities for life cycle management, planning budgeting, acquisition and support of computer resources in Battlefield Automated Systems. For each of these systems utilizing computer resources, a Computer Resources Management Plan (CRMP) shall be prepared by the Acquisition Manager. The CRMP defines the requirements for software development, test and support and assigns responsibilities for the performance of related tasks over the course of a system's life cycle.
9. CECOM CSE has initiated an effort to develop a new acquisition model that addresses the dynamic nature of requirements. This model separates and provides separate contractual efforts for the definition and delineation of requirements that utilizes rapid prototyping and

other technologies from the development effort. The model includes the concept of evolutionary development/block program and will be applied to a new development on a trial basis.

10. AMC has established an intensive two year software engineering intern program.
11. CECOM CSE has established a government/industry documentation task force to examine the documentation process and to develop a framework to reduce documentation cost and improve document utilization.
12. AMC has adopted a version of SECOMO for use in cost estimation
13. CECOM CSE has initiated an effort to develop an improved process model that incorporates the dynamic nature of requirements.
14. CECOM CSE has an effort to develop Ada composite benchmarks for real-time domains and expedite the use of Ada by working with system developers in overcoming the problem being encountered with the application of Ada.
15. CECOM CSE is providing assessments of CASE tools and has initiated the establishment of a CASE showcase.
16. CECOM CSE has initiated an effort to investigate technology for reverse/re-engineering.
17. CECOM CSE has initiated efforts to promote and export Software Reuse technology.
18. CECOM CSE has initiated a study with the National Security Industrial Association to examine the cultural/legal aspects of software reuse and provide recommendations for encouraging software reuse.
19. CECOM has mandated the use of SEI's contractor Software Capability assessment within the selection process for all mission critical software procurements over $10 million.
20. AIRMICS has initiated an effort to aid in migrating current operational non-Ada standard automated systems to Ada.
21. AIRMICS is developing a plan to have all USAISEC-ISSC development centers undergo the SEI assessment process.
22. AIRMICS initiated a project to enhance and improve software maintenance via the use of automated tools.
23. AIRMICS is a member of three NSF centers of excellence to leverage academic/university RDT&E.
24. AIRMICS completed a study on "Incentives for Resue of Ada Components." This study addresses economic and organizational issues in reuse and discusses some case studies in reuse.
25. AIRMICS initiated a study, to evaluate the Distributed Computing Design System to determine its appropriateness in an MIS software engineering support environment.
26. AIRMICS is developing a methodology for the qualitative assessment of software support organizations.
27. AIRMICS completed a study designed to establish guidelines for developing Decision Support Systems and Expert Systems.
28. AIRMICS is sponsoring research into the development of Executive Information Systems to assist the Army Acquisition Executive.
29. AIRMICS is participating in the STEP to recommend software development and test and evaluation criteria. The STEP is sponsored by the Assistant Secretary of the Army (Operations Research) and chaired by the Operational Test and Evaluation Agency.
30. AIRMICS initiated a task to implement a new method for predicting software reliability based on the proportional hazard model, in order to determine operational readiness at major design reviews.

E.3.3 Navy Actions

1. NAVSEA 06 Weapons and Combat Systems Guidance and Policy Paper 88-08, "Software Quality Improvement (SQI) Program", was approved for implementation by NAVSEA field activities on 27 December 1988. It implements techniques to improve the quality control of tactical software and the reporting of development status.

It has been given to two NAVSEA field activities to implement. Each activity has been instructed that the techniques should first act to improve the development process and second act as indicators to management (i.e., NAVSEA) of how the project is progressing. Fleet Combat Direction Systems Support Activity (FCDSSA), Dam Neck has assigned responsibility for it's internal implementation to the Quality Assurance Officer and FCDSSA, San Diego has elected to keep this responsibility within the project management organization. The use of management indicators has been initiated on Advanced Combat Direction Systems (ACDS) Block 0 operational programs for aircraft carrier, amphibious assault, cruiser, and destroyer ship class developments.

High level flag reviews, called for in the instruction, have been implemented at NAVSEA on all programs. These include readiness reviews for Combat System Integration Testing (CSIT) and fleet delivery reviews.

Each activity is presently learning by "trial and error" how the indicators should be implemented. To date, only a few of the eleven planning and progress indicators have been chosen in order to facilitate the institutionalization of the indicators. It is NAVSEA's intent to bring the two activities (who are at the forefront of SQI) together in order to form a consistent common set of indicators that may be used to report progress to NAVSEA.

2. The mission statements for FCDSSA, Dam Neck and FCDSSA, San Diego are set forth in NAVSEAINST 5440.41B of 7 June, 1988, and their stated mission includes: planning, designing, constructing, testing, and delivering Navy tactical computer programs and correcting, updating, modifying, and enhancing these computer programs as required. As Resource Management System (RMS) activities, the FCDSSA's are provided with an Expense Operating Budget (E.O.B.) to execute those assigned tasks included in their mission statements. The E.O.B. funds are programmed annually for the FCDSSA's and issued to them on a quarterly basis. A Ship's Project Directive (SPD) is provided each year to each FCDSSA defining in detail the effort to be accomplished with that year's funding. This SPD is updated throughout the year as changes occur to funding levels and work priorities.
3. NAVSEA, through the FCDSSA's (it's agents for the life-cycle maintenance of Combat Direction Systems (CDS) tactical computer programs) provides for the improvement of the CDS computer programs in deployed units. There are periodic as well as as-required updates to these programs. More extensive baseline and block upgrades to these programs are also provided.
4. An extensive management structure exists in NAVSEA to provide for the configuration management of software and the control of changes. Starting at local change control boards (CCB's) at NAVSEA field activities and working up through the hierarchy of weapon system, program, and platform CCB's, intensive management oversight is provided to the implementation of any proposed software changes.
5. Restructured Naval Tactical Data System (RNTDS) is an operational program production process and task oriented software architecture. The purpose of RNTDS is to facilitate the dual goal of efficiently developing operational programs and providing a common library of reusable software. The Common Reusable Library (CRL) is currently employed in the development of ACDS Block 0 CG/DDG baselines at FCDSSA Dam Neck. The RNTDS production process employs relational database management technology and program segment binding tools to access the CRL and construct operational program tapes. The components of the CRL are written in CMS-2Y Rev. 15.3. The RNTDS architecture is being considered for additional baselines and transition of the CRL to Ada is being investigated. The success of RNTDS is being proven on the CGN 36 production, in which ninety per cent of the operational program is reused software. The CGN 36 will consist of approximately 3000 already tasks/CSRs. Of these 3000 tasks, approximately 2700 already exist in the CRL as a result of previous baseline developments.
6. In response to SECNAVINST 5234.2 and OPNAVINST 5200.28, the Ada Language System/Navy (ALS/N) is currently being developed by NAVSEA PMS 412. ALS/N is

being developed in accordance with MIL-STD-1815A. The ALS/N is the Ada compiler and the minimum programming support environment (MAPSE) developed for the Navy standard computers. ALS/N will allow developers to build Ada programs for deployment in AN UYK-43, AN UYK-44, and AN AYK-14 computers. The ALS/N MAPSE is a critical first step in the Navy's move to Ada for operational programs. The Ada programming language facilitates the development of reusable transportable software more readily than CMS 2.

7. To remain in accordance with OPNAVINST 9410.1, the FCDSSAs perform MULTOTS testing in advance of release of tactical computer programs to NTISA. The MULTOTS are then performed by NTISA prior to joint interoperability testing. Therefore, the FCDSSAAs build toward interoperability through self evaluation at the Navy interoperability level.
8. NAVSEAINST 3560.1A provides for improvements in deployed Navy software for Combat Systems.

E.3.4 Air Force Actions

1. Beginning in Summer 1984, Air Force Systems Command and the Air Force Military Personnel Center cooperated in the establishment of three new Air Force specialty codes for mission-critical computer resource personnel. These three specialty codes (2736, 2885, 2625) allow the identification, education, and tracking of computer systems acquisition managers, engineers, and research scientists, as well as providing a career path to senior positions in the Air Force

   In 1984, the Air Force merged the communications and computer systems communities into a single functional series to parallel the convergence of the technologies. A new Air Force Specialty Code (49xx) was created to describe this group of Command & Control and Management Information Systems support specialists. An aggressive education and cross-training program was implemented to ensure mission support by knowledgeable professionals.
2. The SCC has created a formalized Productivity Group within its organization. The Special Projects Directorate (HQ SSC/PRS) is responsible for managing the implementation of the SEI Action Plan, i.e., recommendations to obtain maximum output for each investment of software resources.
3. Military Airlift Command (MAC) had created the Deputy Chief of Staff for Command, Control, Communication, and Computer Systems to collect and focus resources on software issues.
4. HQ TAC/SC, QPACAF/SC, HQ AAC/SC, and HQ USAFE/SC all have established MAJCOM small computer technical centers. These offices provide a central point of contact and enhance technology gathering.
5. HQ AFLC and HQ AFSC [will soon] publish AFSC/AFLC Pamphlet 800-45, Software Risk Abatement, which provides a set of tools and techniques to assess risk and to enhance software risk abatement.
6. AFR 800-14 requires the incorporation of a risk management plan for computer resources and software in the system risk management plan. It also requires the establishment of the CRWG during the concept exploration phase and the initiation of the CRLCMP. The CRWG and the CRLCMP allow the designated support organization to be involved very early in the system development instead of waiting for FSD.
7. AFSC/AFLC Supplement 1 to AFR 800-14, Sep 87, established the policy that the CRWG will recommend and the program manager (PM) will make an early decision regarding the application of IV&V, and that the support agency will be primary consideration for performing IV&V. IV&V is a successful tool for reducing software errors. AFLC is a strong proponent for IV&V, especially when AFLC is assigned to perform support. This provides an additional benefit of learning the system and reducing the cost of training.
8. In 1986, HQ AFLC allocated a number of high grade nonsupervisory positions to the ALCs; in 1987, PALACE ACQUIRE was implemented. PALACE ACQUIRE is used to

recruit college graduates in the engineering and computer fields.

9. In 1987 HQ AFLC developed the SECCEP which is currently in coordination. SECCEP will provide highly qualified scientists and engineers a career path in the GS13-15 levels. It applies to all scientists and engineers, not just software.
10. The Air Force has mid-career engineering programs through its Air Force Institute of Technology and Air University training programs. DSMC courses can also be used to supplement them.
11. On 15 Jan 88, an AFLC/CC letter authorized the establishment of the software technology support center at OO-ALC. The Center's purpose is to provide a command focus for proactive management of software tools and environments. The center will be responsible for cataloging, prototyping, and supporting tools used to support mission critical computer resources. The center will also manage transition of support tools to other ALCs, establish reusability and supportability requirements, identify near and long-term support requirements, and report AFLC needs to government and industry tool developers.
12. A formal corporate review procedure has been implemented at the SSC as a result of the SEI action plan. This procedure implements a corporate review process to assess the preformance of all SSC development projects. SSC's Command and Control Systems Office at Tinker AFB has a program management review board to provide oversight and review of software development projects.
13. A General Officer/Senior Executive Service level course called "Bold Stroke" was developed to sensitize Air Force senior managers to the critical role that software plays in virtually all of our weapons systems.

E.4 Mapping Tables

Table E-2 shows a mapping between: (1) the problems/issues listed in Section E.2, (2) the recommendations of Section E.3; and (3) the actions taken of Section E.4. There is no one-to-one correlation between the identified recommendations and the actions taken; they are grouped by issue area only. Brief one-line summaries are presented in the tables beside each reference to a recommendation or action as an aid to the reader. It is impossible to capture the full meaning of each recommendation or action in one line and it is therefore suggested that the reader refer back to the full text of the recommendations or actions provided in Sections E.3 and E.4 as necessary.

Table E.2:
E.2.1 Policy

Issues	Recommendation	Action Taken
E.2.1.1 Lack of a defined overall software management, development, and requirements policy	USAF83,#4 Establish AF policy on software risk management USAF83,#6 Establish policy on software oversight management CODSIA84,#2 Modify existing policies (5000.29,31) CODSIA84,#4 Improve CR policy and CR initiative coordination DSB87,#13 Adopt 4-category classification (COTS,Ext.,Emb.,Adv.) DSB87,#14 Develop acquisition policy for each category DSB87,#21 Revise regulations to approach modern commercial practices OIG88,#1 Revise DoD 5000.29 to include minimum requirements for CR plans SEI88a,#44 Alter procurement policies to permit evolutionary devel. methods SEI88a,#46 Establish and use policies on SW envs., metrics, & quality SEI88a,#48 Institute evolutionary development concept ZRAKET88,#1 Revise DoD 5000.29 as soon as possible ZRAKET88,#5 Direct the services to supplement DoD 5000.1 AMC89,#24 Establish clear acquisition policy for software AMC89,#25 Clarify funding policy for software support AFSB89,#16 Establish policy to motivate innovative acquisition tailoring	Army,#1 CECOM *Policy for Mission Critical Defense Systems* issued 7/17/89 Army,#2 CECOM Reg 11-31, *Comp. Resource Management Policy* issued 10/8/88 Army,#3 CECOM Reg 11-25, *Ada Policy* issued 6/1/88 Army,#4 CECOM Reg 700-55, *Replication & Distribution Policies* issued 5/30/89 Navy,#1 NAVSEA 06 Weapons and Combat Systems Guidance and Policy approved Navy,#2 FCDSSAs provided with funding policy AirForce,#5 Published AFSC/AFLC Pamphlet 800-45 AirForce,#6 AFR 800-14 requires risk management plan, CRWG, & CRLCMP AirForce,#12 Formal corporate review procedure implemented at SSC OSD,#8 DoDD 5000.31 replaced by 3405.1 & 3405.2 OSD,#9 DoDD 5000.29 is currently under revision OSD,#15 STARS Program was initiated in November, 1984 OSD,#18 Exec. Working Group est. to review DoD policy on SW systems
E.2.1.2 Policies not current with or conducive to latest technology	DOD82,#1d Develop microprocessor and firmware policies CODSIA84,#2 Modify existing policies (5000.29,31) CODSIA84,#3 Develop a Form-Fit-Interoperability-Transportability (FFIT) policy DSB87,#14 Develop acquisition policy for each category of SW DSB87,#21 Revise regulations to approach modern commercial practices DSB87,#23 Update Directive 5005.29 to incorporate incremental development DSB87,#24 Revise DoD STD 2167 to remove "waterfall" elements DSB87,#25 Mandate risk management in acquisition DSB87,#26 Promote rapid prototyping across the Services OIG88,#1 Revise DoD 5000.29 to include minimum requirements for CR plans SEI88a,#23 Examine DoD2167A and 2168 to locate barriers to reuse SEI88a,#41 Identify technology inhibitors in acquisition practices SEI88a,#44 Alter procurement policies to permit evolutionary devel. methods SEI88a,#46 Establish and use policies on SW envs., metrics, & quality SEI88a,#48 Institute evolutionary development concept ZRAKET88,#1 Revise DoD 5000.29 as soon as possible AFSB89,#1 Include alternatives to waterfall model in forthcoming Handbook 287 AFSB89,#18 Consider revision of AFR 800-14 to require testing each instruction	Navy,#1 NAVSEA 06 Weapons and Combat Systems Guidance and Policy approved OSD,#9 DoDD 5000.29 *is currently under revision* OSD,#11 DoD-HBK 287 reoriented to tailoring guidance
E.2.1.3 Lack of policy to support reuse	SEI88a,#21 Develop policy on responsibilities in reuse library systems SEI88a,#22 Explore policy on software warranties in reusable lib. systems	Navy,#1 NAVSEA 06 Weapons and Combat Systems Guidance and Policy approved

E.2.1.4 Inadequate data rights policy	DSB87,#22 Espouse the concepts of FAR 27.4 for data rights DSB87,#29 Develop economic incentives for developing modules for reuse SEI88a,#20 Develop software data rights for reuse SEI88a,#42 Develop model contracts spanning extremes of data rights SEI88a,#53 Develop limited-rights licensing and depreciation methods for SW ZRAKET88,#9 Issue draft of revised DAR Rights-in-Data Clause	
E.2.1.5 Inadequate PDSS and software maintenance policy	USAF83,#13 Standardize and support PDDS environments SEI88a,#48 Institute evolutionary development concept IDA89,#1 Establish approach for improving deployed military SW	Army,#22 AIRMICS initiated project to automate software maintenance Navy,#3 FCDSSAs provide life-cycle support Navy,#8 NAVSEAINST 3560.1A provides for improvements in deployed SW
E.2.1.6 Inadequate enforcement of policies	DSB87,#5 Commit DoD management to a serious and determined push to Ada BOARD88,#2 DoD Directive 3405.2 should be strongly enforced ZRAKET88,#2 Provide regularly-updated handbooks to aid implementing 2167A AMC89,#33 Address software as part of materiel release AMC89,#36 Enforce standard software cost model use AFSB89,#20 Expedite, distribute, and maintain the 2168 guidebook	Navy,#6 ALS/N is under development OSD,#3 DoD Directives 3405.1, 3405.2 OSD,#9 DoDD 5000.29 is currently under revision OSD,#16 2168 Guidebook published as MIL-HDBK-286
E.2.1.7 Software policies uncoordinated and in conflict with one another	CODSIA84,#4 Improve CR policy and CR initiative coordination CODSIA84,#14 Dev. tech. FFIT policy of strategy, planning, and standardization OIG83,#2 Revise DoD5000.29 and 5000.39 to reference each other OIG83,#7 Define plan to periodically review and coordinate waivers to using Ada	OSD,#3 DoD Directives 3405.1, 3405.2
E.2.1.8 Lack of appropriate standards	CODSIA84,#1 Optimize DoD use of computer resources with high-level standards CODSIA84,#7 Update hardware standards and ISA policy CODSIA84,#11 Raise the level of standards and make them tailorable CODSIA84,#13 Establish team (industry, government, academia) to address CR issues DSB87,#16 Investigate methods for selection and standardization of COTS	
E.2.1.9 No national strategy for the software engineering field	CODSIA84,#1 Optimize DoD use of computer resources with high-level standards CODSIA84,#14 Dev. tech. FFIT policy of strategy, planning, and standardization CODSIA84,#15 Dev. management concept of Flexible Application of Standards & Tech. DSB87,#2 Task STARS,AJPO,SEI,DARPA, and SDI to produce a joint plan SEI88a,#4 Develop vision of nation's direction in software capability SEI88a,#5 Identify person(s) to form a national strategy for software ZRAKET88,#13 Create a unified DoD Software Technology Plan	OSD,#2 DAB Science & Technology WG formed to define SW Master Plan

Table E.2

E.2.2 Management

E.2.2.1 Management Methods

Issues	Recommendations	Action Taken
E.2.2.1.1 Inadequate software management concepts, methods, practices	CODSIA84,#15 Dev. management concept of Flexible Application of Standards & Tech. SEI88a,#16 Clarify the government oversight role SEI88a,#18 Continuously update the Software Development Plan SEI88a,#19 Improve overall role and guidance by Government SEI88a,#48 Institute evolutionary development concept AMC89,#26 Provide for management of software change	Army,#5 *CECOM CSE Master Plan* [in preparation] Navy,#4 NAVSEA Change Control Boards provide configuration management OSD,#10 DoD STD-2167A has been issued
E.2.2.1.2 Lack of management attention/commitment to software issues	OIG88,#4 Require Service counterparts of DAB to review CR plans SEI88a,#16 Clarify the government oversight role SEI88a,#39 Provide government oversight as costed and scheduled item AFSB89,#10 Monitor and evaluate CRWG performance	Army,#2 CECOM Reg 11-31, *Comp. Resource Management Policy*, issued 10/8/89 OSD,#9 DoDD 5000.29 is currently under revision
E.2.2.1.3 Lack of commitment to Ada	DSB87,#5 Commit DoD management to a serious and determined push to Ada DSB87,#9 Increase investment in Ada education and training BOARD88,#2 *DoD Directive 3405.2 should be strongly enforced* OIG88,#6 Require annual updates of Ada implementation plans by Services OIG88,#7 Define plan to periodically review waivers to using Ada AMC89,#6 Manage the introduction of Ada into the Army	Army,#3 CECOM Regulation 11-25, Ada Policy, issued 6/1/88 Army,#6 CECOM policy for providing Ada compilers to univs. issued 7/29/89 Army,#20 AIRMICS initiated effort to migrate current systems to Ada OSD,#3 DoD Directives 3405.1, 3405.2 OSD,#19 ATIP established
E.2.2.1.4 Lack of adequate planning and focus on front-end processes of life-cycle	AMC89,#39 Provide efficient front end loading	
E.2.2.1.5 Inadequate risk assessment	USAF83,#4 Establish AF policy on software risk management USAF83,#5 Provide risk management information support DSB87,#12 Use evolutionary acquisition to reduce risk DSB87,#25 Mandate risk management in acquisition SEI88a,#47 Implement programs to analyze overall software process AFSB89,#2 *Ensure adequate risk reduction prior to full-scale development* AFSB89,#4 Mandate early identification of critical software issues	OSD,#9 DoDD 5000.29 is currently under revision OSD,#15 STARS program was initiated in November, 1984
E.2.2.1.6 Poor management methods for software support	AMC89,#34 Develop responsive distribution process AMC89,#40 *Consider alternative support options (via contractors)*	Army,#26 AIRMICS developing methodology for assessing SW support orgs. Navy,#8 NAVSEAINST 3560.1A provides for improvements in deployed SW

E.2.2.1.7 Software planning not integrated with system planning	DOD82,#2a Develop tools to assess impact of proposed system changes AMC89,#9 Conduct integrated software planning AMC89,#32 Implement effective interoperability control AMC89,#37 Improve interface into PPBS for software	Navy,#7 FCDSSAs perform interoperability testing
E.2.2.1.8 Inadequate metrics and monitoring tools	DOD82,#1a Develop better monitoring approaches and tools DSB87,#19 Develop and use metrics for software quality & completeness DSB87,#20 Develop metrics for measuring implementation progress SEI88a,#17 Improve criteria for selecting management indicators SEI88a,#47 Implement programs to analyze overall software process SEI88a,#50 Develop ability to measure weaknesses in software process AMC89,#3 Develop and evaluate software metrics AMC89,#35 Provide software maturity management with use of metrics AFSB89,#22 Select key programs to demonstrate prototype quality assessment	Army,#7 CECOM CSE effort in process metrics Navy,#1 NAVSEA 06 Weapons and Combat Systems Guidance and Policy approved OSD,#4 AFSCP 800-14, 800-43; AMC P 70-13, 70-14; DoD 5000.3-M-3 OSD,#5 AFSCP 800-43; DoD 5000.3-M-3 OSD,#15 STARS program was initiated in November, 1984

Table E.2

E.2.2 Management

E.2.2.2 DoD Management Structure

Issues	Recommendations	Action Taken
E.2.2.2.1 Efforts are uncoordinated	DOD82,#2g Promote greater coordination among Service elements USAF83,#3 Establish computer technology center for AF computer issues USAF83,#15 Establish single authority to assure post-deployment integrity CODSIA84,#10 Integrate current initiatives and add system-level initiative DSB87,#1 Create a Joint Program Office for STARS, AJPO, and the SEI DSB87,#2 Task STARS,AJPO,SEI,DARPA, and SDI to produce a joint plan DSB87,#6 Move AJPO into same organization as STARS and SEI BOARD88,#2 DoD Directive 3405.2 should be strongly enforced AMC89,#20 Provide virtual colocation with TRADOC centers via comm. network AMC89,#29 Enhance interaction between activities IDA89,#3 Establish lead agents for selected critical software technology areas	AirForce,#2 Established SCC Formal Productivity Group AirForce,#3 Created MAC Deputy Chief of Staff for C3 and Computer Systems AirForce,#4 Established MAJCOM small computer technical centers AirForce,#6 AFR 800-14 requires risk management plan, CRWG, & CRLCMP OSD,#1 OSD moved STARS from AF; STARS & SEI now together in DARPA OSD,#2 DAB Science & Technology WG formed to define SW Master Plan OSD,#18 Exec. Working Group est. to review DoD policy on SW systems
E.2.2.2.2 Lack of clearly defined roles and responsibilities	USAF83,#6 Establish policy on software oversight management DSB87,#2 Task STARS,AJPO,SEI,DARPA, and SDI to produce a joint plan DSB87,#28 Define the role of Using Commands in evolutionary development SEI88a,#19 Improve overall role and guidance by Government AMC89,#14 Establish clear organizational responsibilities IDA89,#4 Consolidate DoD computer policy, management, and oversight	Army,#8 AMC-Reg 70-16A [re policies for Battlefield Automated Sys.] issued 8/16/89 AirForce,#12 Formal corporate review procedure implemented at SSC OSD,#2 DAB Science & Technology WG formed to define SW Master Plan OSD,#18 Exec Working Group est. to review DoD policy on SW systems

E.2.2.3 Structure is inadequate	CODSIA84,#12 Formalize & streamline the CR organizational structure within DoD DSB87,#1 Create a Joint Program Office for STARS, AJPO, and the SEI DSB87,#6 Move AJPO into same organization as STARS and SEI BOARD88,#1 AJPO should continue to function at OSD level OIG88,#3 Appoint an OSD office as technical advisor to DAB SEI88a,#5 Identify person(s) to form a national strategy for software AMC89,#12 Organize Army to manage acquisition process AMC89,#13 Improve PM/PEO and MACOM interface AMC89,#15 Strengthen AMC's software management role AMC89,#16 Provide one-stop support through life-cycle SE center AMC89,#17 Build an Army software tech. center with SE intern program AMC89,#19 Provide transition for support for training device AMC89,#27 Establish internal controls and feedback AMC89,#42 Measure efficiency of current LCSE centers IDA89,#4 Consolidate DoD computer policy, management, and oversight	OSD,#1 OSD moved STARS from AF; STARS & SEI now together in DARPA OSD,#18 Exec. Working Group est. to review DoD policy on SW systems
E.2.2.2.4 No structural link between deployed software needs and the software Research and Development community	DSB81,#2 Develop investment strategy to link investment and technical needs DSB87,#11 Focus research on SDI-specific objectives IDA89,#1 Establish approach for improving deployed military SW	

Table E.2

E.2.3 Life Cycle Management

E.2.3.1 Requirements

Issues	Recommendations	Action Taken
E.2.3.1.1 Requirements are ill-defined, incorrect, and/or do not meet user's needs	DSB87,#12 Use evolutionary acquisition to reduce risk DSB87,#23 Update Directive 5005.29 to incorporate incremental development DSB87,#24 Revise DoD STD 2167 to remove "waterfall" elements DSB87,#26 Promote rapid prototyping across the Services SEI88a,#43 Use evolutionary development method SEI88a,#48 Institute evolutionary development concept ZRAKET88,#7a Reduce risk via improved requirements definition AMC89,#4 Evaluate software life cycle models AMC89,#10 Tailor SW acq. process rather than rote use of stds. AMC89,#39 Provide efficient front end loading AFSB89,#6 Tailor user involvement for each program	Army,#9 CECOM CSE initiated effort to develop new acquisition model 8/89 Army,#11 CECOM CSE established documentation task force 7/89 OSD,#9 DoDD 5000.29 is currently under revision

E.2.3.1.2 Requirements undergo significant uncontrolled change	DOD82,#1b Develop life-cycle model that reflects changable requirements AFSB89,#5 Support deferral of late SW requirements	Army,#9 CECOM CSE effort to develop new acquisition model 8/89 Army,#11 CECOM CSE established documentation task force 7/89 OSD,#9 DoDD 5000.29 is currently under revision
E.2.3.1.3 Requirements lack system view	DOD82,#2a Develop tools to assess impact of proposed system changes SEI88a,#48 Institute evolutionary development concept and systems focus AMC89,#9 Conduct integrated software planning AMC89,#37 Improve interface into PPBS for software	OSD,#9 DoDD 5000.29 is currently under revision
E.2.3.1.4 Inadequate documentation and allocation/traceability of requirements	DOD82,#2e Support improved documentation practices	OSD,#9 DoDD 5000.29 is currently under revision
E.2.3.1.5 No capability to perform cost/benefit analysis of requirements change	DOD82,#2a Develop tools to assess impact of proposed system changes	

Table E.2

E.2.3 Life Cycle Management

E.2.3.2 Costing/Forecasting

Issues	Recommendations	Action Taken
E.2.3.2.1 Lack of resource forecasting model for new methods/technologies	DOD82,#2c Develop metrics for cost, productivity, and quality AMC89,#36 Enforce standard software cost model use	Army,#12 AMC Adopted version of SECOMO Navy,#1 NAVSEA 06 *Weapons and Combat Systems Guidance and Policy* approved OSD,#13 AJPO funded study (showed std cost model poor idea)
E.2.3.2.2 Lack of historical basis for predicting/estimating software	OIG88,#5 Consider using individual databases to track defense software AMC89,#28 Develop a computer resource database AMC89,#36 Enforce standard software cost model use AMC89,#38 Identify and capture actual software cost	

Table E.2

E.2.3 Life Cycle Management

E.2.3.3 Acquisition

Issues	Recommendations	Action Taken
E.2.3.3.1 Software capabilities not adequately addressed in system contracting	AMC89,#10 Tailor SW acq. process rather than rote use of standards AMC89,#11 Develop a consistent contracting approach addressing SW	
E.2.3.3.2 Current acquisition process inhibits innovative methods	USAF83,#9 Encourage technology transfer in acquisition practices SEI88a,#15 Incorporate "SW Contract Evaluator" into acquisition process SEI88a,#38 Encourage interaction and feedback with SW contractor SEI88a,#40 Use (European) competitive reimbursement contract method SEI88a,#41 Identify technology inhibitors in acquisition practices AMC89,#10 Tailor SW acq. process rather than rote use of stds. AFSB89,#16 Establish policy to motivate innovative acquisition tailoring	Army,#28 AIRMICS research in developing Executive Information Systems
E.2.3.3.3 Difficult under current contracting mechanisms for contractors to capitalize software assets	SEI88a,#49 Develop market for investment in software SEI88a,#53 Develop limited-rights licensing and depreciation methods for SW ZRAKET88,#3 Direct the Services to create SW productivity line item	
E.2.3.3.4 Inadequate education with respect to innovative acquisition options	SEI88a,#13 Develop model SOW clauses geared to using new technologies SEI88a,#42 Develop model contracts spanning extremes of data rights AFSB89,#7 Tailor contract form for each program's needs	
E.2.3.3.5 Lack of productivity and quality incentives	DOD82,#2c Develop metrics for cost, productivity, and quality DSB87,#17 Develop productivity incentives for custom-built software DSB87,#18 Develop increased profit incentives on software quality DSB87,#19 Develop and use metrics for software quality & completeness SEI88a,#45 Emphasize higher quality software products SEI88a,#46 Establish and use policies on SW envs., metrics, & quality SEI88a,#52 Require strong error prevention throughout life cycle ZRAKET88,#4 Develop new contract specs for reuse/quality/productivity AMC89,#30 Integrate software quality into process	Navy,#1 NAVSEA 06 Weapons and Combat Systems Guidance and Policy approved OSD,#4 AFSCP 800-14, 800-43; AMC P 70-13, 70-14; DoD 5000.3-M-3

Table E.2
E.2.3 Life Cycle Management
E.2.3.4 Design/Analysis

Issues	Recommendations	Action Taken
E.2.3.4.1 Software product designs become unwieldy for very complex, high performance, and changing requirements	SEI88a,#43 Use evolutionary development method SEI88a,#48 Institute evolutionary development concept ZRAKET88,#7b Reducing risk via incremental implementation ZRAKET88,#7c Schedule preliminary design review at appropriate time	Army,#9 CECOM CSE initiated effort to develop new acquisition model 8/89 Army,#11 CECOM CSE established documentation task force 7/89 OSD,#9 DoDD 5000.29 is currently under revision OSD,#15 STARS Program was initiated in November, 1984 OSD,#20 DAB approved incremental development for WAM and DMS
E.2.3.4.2 Mechanism lacking for early collaboration to establish and resolve common interfaces	SEI88a,#43 Use evolutionary development method AFSB89,#9 Est. management criteria to handle physically separate design groups	OSD,#9 DoDD 5000.29 is currently under revision

Table E.2
E.2.3 Life Cycle Management
E.2.3.5 Development Methodology

Issues	Recommendations	Action Taken
E.2.3.5.1 Waterfall model is usually inadequate, but alternative models are not fully developed and must address complex DoD application domain	DOD82,#1b Develop life-cycle model that reflects changable requirements DOD82,#2b Sponsor research in design of state-of-the-art systems USAF83,#11 Investigate methodology to ensure high integrity software DSB87,#24 Revise DoD STD 2167 to remove "waterfall" elements SEI88a,#43 Use evolutionary development method SEI88a,#48 Institute evolutionary development concept AMC89,#4 Evaluate software life cycle models and improve specification process AMC89,#10 Tailor SW acq. process rather than rote use of stds. AFSB89,#1 Include alternatives to waterfall model in forthcoming Handbook 287	Army,#9 CECOM CSE initiated effort to develop new acquisition model 8/89 Army,#11 CECOM CSE established documentation task force 7/89 Army,#13 CECOM CSE effort to develop improved process model
E.2.3.5.2 Methodologies for evaluating development methods generally do not exist	SEI88a,#50 Develop ability to measure weaknesses in software process ZRAKET88,#6 RFPs should require defined software development processes AMC89,#4 Evaluate software life cycle models and improve specification process	OSD,#6 Front-end risk reduction addressed by DoD 5000.3, 5000.3-M-3, 2167A OSD,#7 Baselines, designs, rqts, tests addressed in MIL-STD 483A, 1521B

Table E.2
E.2.3 Life Cycle Management
E.2.3.6 Languages

Issues	Recommendations	Action Taken
E.2.3.6.1 Inability to calibrate tradeoffs of language performance vs. downstream system capabilities	OIG88,#5 Consider using individual databases to track defense software AMC89,#3 Establish framework and eval. product, process, and compiler metrics AMC89,#6 Manage the introduction of Ada into the Army	Navy,#1 NAVSEA 06 Weapons and Combat Systems Guidance and Policy approved
E.2.3.6.2 Need to integrate 4th/5th generation languages into DoD mainstream	DSB87,#10 Allow 4th-generation languages when savings are tenfold	
E.2.3.6.3 Ability to obtain predictable real-time performance with Ada has not been demonstrated	CODSIA84,#9 Expand run-time tech. activities toward common run-time interfaces DSB87,#4 Direct STARS to build upon existing real programs AMC89,#3 Establish framework and eval. product, process, and compiler metrics AMC89,#6 Manage the introduction of Ada into the Army	Army,#3 CECOM Reg 11-25, *Ada Policy* issued 6/1/88 Army,#14 CECOM CSE effort to develop composite Ada benchmarks Navy,#1 NAVSEA 06 Weapons and Combat Systems Guidance and Policy approved OSD,#21 ACEC test suite addresses Ada performance issues
E.2.3.6.4 No widespread implementation of Ada chapter 13		OSD,#14 Latest ACVC (1.11) includes Ch 13 tests OSD,#22 Ada 9X effort is revising the Ada standard

Table E.2
E.2.3 Life Cycle Management
E.2.3.7 Development and Support Tools/Environments

Issues	Recommendations	Action Taken
E.2.3.7.1 Insufficient tech base hampers ability to use tools	CODSIA84,#8 Continue efforts in APSE and CAIS SEI88a,#54 Develop tools and methods for rapid prototyping ZRAKET88,#12 Continue current near-/long-term efforts in environments	OSD,#15 STARS Program was initiated in November, 1984

E.2.3.7.2 Lack of consensus on the use of methods and tools results in their inconsistent application	SEI88a,#46 Establish and use policies on SW envs., metrics, & quality AMC89,#1 Develop commercial product testbed, tech R&D, and focus on reuse AMC89,#5 Establish reqs. and stds. for extensible software environments AFSB89,#8 Require use of a SEE for all programs larger than 12 people AFSB89,#25 Consider funding an evaluation of Software Engineering Environments	Army,#15 CECOM CSE providing CASE tool assessments Army,#27 AIRMICS study to establish guidelines for developing Expert Systems Navy,#1 NAVSEA 06 Weapons and Combat Systems Guidance and Policy approved OSD,#15 STARS Program was initiated in November, 1984 OSD,#17 SEI tasked to evaluate SEEs
E.2.3.7.3 Tools do not support the full life cycle		
E.2.3.7.4 Tools do not integrate or port	DOD82,#2d Develop integrated environments AMC89,#5 Establish reqs. and stds. for extensible software environments	Army,#25 AIRMICS study on evaluation of Distributed Computing Design System OSD,#15 STARS Program was initiated in November, 1984
E.2.3.7.5 Lack of tools that support parallel and distributed systems, particularly runtime instrumentation	DOD82,#2b Sponsor research in design of state-of-the-art systems	
E.2.3.7.6 DoD has a heavy bias toward custom-built methods, tools, and software	DSB87,#15 Direct system managers to assume COTS systems will suffice DSB87,#16 Investigate methods for selection and standardization of COTS	OSD,#18 Exec. Working Group est. to review DoD policy on SW systems
E.2.3.7.7 Adequate support environment must be planned for and acquired during development	USAF83,#13 Standardize and support PDDS environments AMC89,#5 Establish reqs. and stds. for extensible software environments AFSB89,#8 Require use of a SEE for all programs larger than 12 people	OSD,#23 Common APSE Interface Set (MIL-STD 1838-A) developed
E.2.3.7.8 Ineffective software configuration management tools	AMC89,#31 Improve software configuration management	
E.2.3.7.9 Inadequate tools to collect data needed for software measurement	SEI88a,#47 Implement programs to analyze overall software process AMC89,#3 Establish framework and eval. product, process, and compiler metrics	Army,#15 CECOM CSE providing CASE tool assessments Navy,#1 NAVSEA 06 Weapons and Combat Systems Guidance and Policy approved OSD,#15 STARS Program was initiated in November, 1984

Table E.2
E.2.3 Life Cycle Management
E.2.3.8 Test and Evaluation

Issues	Recommendations	Action Taken
E.2.3.8.1 Inadequate verifiability, suitability, and predictive capability of existing testing methods		Army,#29,#AIRMICSparticipating in STEP Program to recommend SW T&E criteria
E.2.3.8.2 Need to fully integrate testing and evaluation into development activities	SEI88a,#52 Require strong error prevention throughout life cycle AMC89,#30 Integrate software quality into process	
E.2.3.8.3 Failure to apply existing methods	USAF83,#10 Increase use of IV&V DSB87,#27 Provide Using Commands with OT&E and change eval. facilities SEI88a,#27 Develop minimal testing standards for reusable components SEI88a,#28 Develop standards to document testing of reusable components ZRAKET88,#11 Investigate technical issues of validating reusable SW *AFSB89,#18 Consider revision of AFR 800-14 to require testing* each instruction AFSB89,#19 Consider using designated maintainer as IV&V agent AFSB89,#21 Require COTS technology in development	AirForce,#6 AFR 800-14 requires risk management plan, CRWG, & CRLCMP AirForce,#7 AFSC/AFLC Supplement 1 to AFR 800-14 issued 9/1/87
E.2.3.8.4 Product assurance groups lack leverage		

Table E.2
E.2.3 Life Cycle Management
E.2.3.9 Adaptive, Corrective, and Preventitive Maintenance

Issues	Recommendations	Action Taken
E.2.3.9.1 No ability to develop complex software that operates correctly and is maintainable	USAF83,#14 Enable early participation by designated support organization	AirForce,#10 AFR 800-14 published 9/29/86

E.2.3.9.2 Software complexity tends to grow during development and with evolutionary upgrades		
E.2.3.9.3 No link between developers and maintainers	AMC89,#19 Provide transition for support for training device AFSB89,#19 Consider using designated maintainer as IV&V agent	AirForce,#6 AFR 800-14 *requires risk management* plan, CRWG, & CRLCMP
E.2.3.9.4 Lack of technology for support of deployed systems, including support of low quality software	AMC89,#7 Establish mechanism for reverse engineering IDA89,#1 Establish approach for improving deployed military SW AFSB89,#13 Colocate support for SW in integrated systems	Army,#4 CECOM Reg. 700-55, *Replication & Distribution Policies* issued 5/30/89 Army,#16 CECOM CSE effort to investigate reverse/re-engineering

Table E.2

E.2.3 Life Cycle Management

E.2.3.10 Modernization/Upgrades

Issues	Recommendations	Action Taken
E.2.3.10.1 Lack of techniques for developing evolvable systems	DOD82,#2f Develop techniques for dealing with rapid technology changes SEI88a,#48 Alter government policies to permit/force evolutionary methods AMC89,#7 Establish mechanism for reverse engineering IDA89,#1 Establish approach for improving deployed military SW	OSD,#20 DAB approved incremental development for WAM and DMS

Table E.2
E 2.4 Product
E.2.4.1 Fullfillment of Requirements

Issues	Recommendations	Action Taken
E.2.4.1.1 Lack of generally accepted and understood acceptance criteria		
E.2.4.1.2 System not interoperable	CODSIA84,#5 Ensure compatibility, use of networks, and architectural innovation CODSIA84,#6 Use Ada for transportability, reusability, and interoperability focus CODSIA84,#11 Raise the level of standards and make them tailorable AMC89,#32 Implement effective interoperability control	OSD,#24 DoD established policies regarding use of DDN and GOSIP
E.2.4.1.3 Quality of software degrades disproportionately with the size and complexity of software	DSB87,#18 Develop increased profit incentives on software quality SEI88a,#45 Emphasize higher quality software products	

Table E.2
E.2.4 Product
E.2.4.2 Reliability/Fault Tolerance

Issues	Recommendations	Action Taken
E.2.4.2.1 Systems are not reliable and do not recover adequately from processor faults		
E.2.4.2.2 There is not an adequate concept of and theory of software reliability		Army,#30 AIRMICS implementing new method for predicting SW reliability

Table E.2

E.2.4 Product

E.2.4.5 Quality Measurement

Issues	**Recommendations**	**Action Taken**
E.2.4.5.1 Product quality metrics are deficient	DOD82,#2c Develop metrics for cost, productivity, and quality DSB87,#19 Develop and use metrics for software quality & completeness SEI88a,#25 Develop metrics to measure reusability of components SEI88a,#47 Implement programs to analyze overall software process AMC89,#3 Establish framework and eval. product, process, and compiler metrics AMC89,#35 Provide software maturity management with use of metrics	Navy,#1 NAVSEA 06 Weapons and Combat Systems Guidance and Policy approved OSD,#4 AFSCP 800-14, 800-43; AMC P 70-13, 70-14; DoD 5000.3-M-3
E.2.4.5.2 Causal relationships among program management process, software development process, and product quality are not proven	SEI88a,#50 Develop ability to measure weaknesses in software process AMC89,#3 Establish framework and eval. product, process, and compiler metrics AFSB89,#22 Select key programs to demonstrate prototype quality assessment	Navy,#1 NAVSEA 06 Weapons and Combat Systems Guidance and Policy approved

Table E.2
E.2.5 Software Reuse and COTS Software
E.2.5.1 Technology

Issues	Recommendations	Action Taken
E.2.5.1 There is no adequate concept or methodology for software reuse	SEI88a,#24 Develop coding standards for reuse library components SEI88a,#25 Develop metrics to measure reusability of components SEI88a,#26 Develop documentation standards for reusable components SEI88a,#27 Develop minimal testing standards for reusable components SEI88a,#28 Develop standards to document testing of reusable components SEI88a,#29 Develop tools to analyze high-payoff domains for reuse SEI88a,#30 Develop cataloging system and tools for reusable SW library SEI88a,#31 Develop retrieval system for reusable software library SEI88a,#32 Choose strategy for library of reusable weapon system SW SEI88a,#33 Determine policy for certification of reusable components SEI88a,#34 Determine policy for access to reuse library SEI88a,#35 Determine system for handling classified reusable components SEI88a,#36 Determine change policy for reuse library AMC89,#1 Develop commercial product testbed, tech R&D, and focus on reuse AMC89,#2 Develop plan, R&D, and policy for software reuse	Army,#17 CECOM CSE effort to promote/export reuse technology Army,#24 AIRMICS study "Incentives for Reuse of Ada Components"

Table E.2
F.2.5 Software Reuse and COTS Software
E.2.5.2 Incentives

Issues	Recommendations	Action Taken
E.2.5.2.1 Incentives and policies are lacking to encourage use of commercial off-the-shelf (COTS) software & reuse	DOD82,#2c Develop metrics for cost, productivity, and quality DSB87,#13 Adopt 4-category classification (COTS,Ext.,Emb.,Adv.) DSB87,#15 Direct system managers to assume COTS systems will suffice DSB87,#29 Develop economic incentives for reuse of DoD modules DSB87,#30 Develop economic incentives for software reuse DSB87,#31 Promote identification of areas of potential reuse SEI88a,#37 Determine incentives for reuse ZRAKET88,#4 Develop new contract specs for reuse/quality/productivity ZRAKET88,#7b Reduce risk via incremental implementation AMC89,#2 Develop plan, R&D, and policy for software reuse AFSB89,#21 Require COTS technology in development	Army,#18 CECOM/NSIA study re cultural/legal aspects of reuse OSD,#25 DoD Directive 5000.37 mandates use of GOSIP

Table E.2
E.2.5 Software Reuse and COTS Software
E.2.5.3 Access/Retrieval

Issues	Recommendations	Action Taken
E.2.5.3.1 There are no domain-specific libraries for reusable components	DSB87,#31 Promote identification of areas of potential reuse DSB87,#32 Have SEI establish a prototype module market DSB87,#33 Have SEI establish standards of description for reusable Ada modules SEI88a,#29 Develop tools to analyze high-payoff domains for reuse SEI88a,#32 Choose strategy for library of reusable weapon system SW ZRAKET88,#11 Investigate technical issues of validating reusable SW AMC89,#2 Develop plan, R&D, and policy for software reuse	OSD,#26 Development of CAMP, GRACE, and Booch components
E.2.5.3.2 There are inadequate means to collaborate with commercial industry to develop COTS components that address defense requirements	DSB87,#31 Promote identification of areas of potential reuse DSB87,#32 Have SEI establish a prototype module market DSB87,#33 Have SEI establish standards of description for reusable Ada modules SEI88a,#37 Determine incentives for reuse ZRAKET88,#4 Develop new contract specs for reuse/quality/productivity	OSD,#15 STARS Program was initiated in November, 1984 OSD,#26 Development of CAMP, GRACE, and Booch components

Table E.2
E.2.6 Personnel
E.2.6.1 Management Personnel

Issues	Recommendations	Action Taken
E.2.6.1.1 Upper management lack of familiarity with software	OIG88,#4 Require Service counterparts of DAB to review CR plans SEI88a,#1 Create briefing on modern SW issues for gov't/industry leaders SEI88a,#2 Inform executives about benefits of software flexibility/functionality SEI88a,#3 Define the implications of modular systems develop. and inform execs. SEI88a,#56 Develop briefing for managers on quantitative SW management AMC89,#21 Develop pilot software awareness program AMC89,#22 Develop operational software literacy program AMC89,#23 Find Army software advocates	Army,#28 AIRMICS research in developing Executive Information Systems OSD,#9 DoDD 5000.29 is currently under revision

E.2.6.1.2 Software Managers lack expertise/education in software engineering	USAF83,#2 Evolve to a DCS-level manager of AF information resources DSB87,#9 Increase investment in Ada education and training DSB87,#37 Encourage officer career paths in technical management SEI88a,#55 Create concept list usable by managers to make SW decisions SEI88a,#56 Develop briefing for managers on quantitative SW management ZRAKET88,#8 Develop 1-day course in innovative SW management AFSB89,#14 Broaden base of personnel skilled in acquisition of software systems	Army,#21 AIRMICS developing plan to use SEI assessment on ISAISEC-ISSC centers OSD,#27 DoD Computer Institute developing course tailored to program managers
E.2.6.1.3 Poor utilization and allocation of software professionals	DSB87,#36 Establish comprehensive personnel database DSB87,#37 Encourage officer career paths in technical management AMC89,#16 Provide one-stop support through life-cycle SE center AMC89,#41 Conduct contracting out study for ADP/MIS systems AFSB89,#14 Broaden base of personnel skilled in acquisition of software systems AFSB89,#15 Create a software systems advisory team with special career tailoring AFSB89,#26 Capture SE knowledge of experts for use in knowledge-based tools	

Table E.2

E.2.6 Personnel

E.2.6.2 Acquisition Expertise

Issues	Recommendations	Action Taken
E.2.6.2 Acquisition people have inadequate knowledge of software	DSB87,#35 Revise personnel strategy in favor of acquisition personnel SEI88a,#13 Develop model SOW clauses geared to using new technologies SEI88a,#14 Develop acceptable, consistent definitions of common terms SEI88a,#15 Incorporate "SW Contract Evaluator" into acquisition process AMC89,#46 Provide job challenge for software engineers AFSB89,#14 Broaden base of personnel skilled in acquisition of software systems	Army,#19 CECOM mandated using SEI's Contractor Assessment 7/14/89

Table E.2

E.2.6 Personnel

E.2.6.3 Real World/Large-Scale System Expertise

Issues	Recommendations	Action Taken
E.2.6.3.1 Academic software engineering courses don't provide adequate experience with large-scale systems	SEI88a,#8 Identify educational roles of government, industry, academia SEI88a,#11 Develop draft program for on-the-job training SEI88a,#12 Develop 5-year co-op education curricula in SE	

Table E.2

E.2.6 Personnel

E.2.6.4 Ongoing and Refresher Training

Issues	Recommendations	Action Taken
E.2.6.4.1 Failure to educate for changing and/or immature *technology*	DSB87,#9 Increase investment in Ada education and training DSB87,#38 Enhance education for software personnel SEI88a,#8 Identify educational roles of government, industry, academia SEI88a,#10 Encourage managers to use retraining strategies SEI88a,#11 Develop draft program for on-the-job training AMC89,#6 Manage the introduction of Ada into the Army AMC89,#44 Improve incentives for military software experts AMC89,#46 Provide job challenge for software engineers AFSB89,#17 Create mid-career systems/software engineering programs	Army,#10 AMC established software engineering intern program 9/86 AirForce,#10 Already implemented (AFIT, Air University training progs).
E.2.6.4.2 Need to improve individual productivity	DOD82,#3c Establish productivity measures DSB87,#36 Establish comprehensive personnel database	

Table E.2
E.2.6 Personnel
E.2.6.5 Retention/Recruitment

Issues	Recommendations	Action Taken
E.2.6.5.1 Shortage of people with specialized skills	DSB81,#4 Provide graduate fellowships tied to required work in DoD labs USAF83,#2 Evolve to a DCS-level manager of AF information resources USAF83,#12 Upgrade personnel for Post-Deployment System Support (PDDS) DSB87,#9 Increase investment in Ada education and training DSB87,#38 Enhance education for software personnel SEI88a,#7 Develop national strategy for recruitment into SW field SEI88a,#9 Create briefing to recruit students into SW profession	AirForce,#8 PALACE ACQUIRE program implemented 1987 AirForce,#9 SECCEP developed by AFLC (1987)
E.2.6.5.3 Lack of career paths	DOD82,#3a Define and advocate systems/software engineering career fields DOD82,#3b Initiate program to exchange government and industry personnel USAF83,#1 Establish high-priority career path for software personnel DSB87,#37 Encourage officer career paths in technical management SEI88a,#9 Create briefing to recruit students into SW profession AMC89,#18 Organize to grow software engineers AMC89,#43 Develop software engineering career program AMC89,#44 Improve incentives for military software experts AMC89,#45 Establish career subprogram *management* AFSB89,#15 Create a software systems advisory team with special career tailoring	AirForce,#1 Established 3 AFSCs (2736,2885,2625) for MCCR personnel

Table E.2
E.2.6 Technology Transition

Issues	Recommendations	Action Taken
E.2.7.1 There is no strategy for dealing with rapidly changing technology and engineering practice	CODSIA84,#2 Modify existing policies (5000.29,31) SEI88a,#6 Define roles of gov't, industry, and academia in technology transition AMC89,#8 Develop a strategy for technology insertion AFSB89,#3 Fund parallel programs at system level to foster wider industrial base AFSB89,#23 Initiate program to foster SE technology transfer	
E.2.7.2 There is a lack of awareness of existing technology	DOD82,#2h Develop framework for technology evaluation and infusion CODSIA84,#3 Develop a Form-Fit-Interoperability-Transportability (FFIT) policy SEI88a,#6 Define roles of gov't, industry, and academia in technology transition AMC89,#8 Develop a strategy for technology insertion	
E.2.7.3 Need for technology demonstration projects	DSB87,#3 Task STARS to define new goals and milestones AMC89,#8 Develop a strategy for technology insertion AFSB89,#12 Select a test program to evaluate benefits of incremental acquisition AFSB89,#22 Select key programs to demonstrate prototype quality assessment AFSB89,#24 Increase technology base investment in software engineering	OSD,#19 ATIP established

Table E.2
E.2.8 Security

Issues	Recommendations	Action Taken
E.2.8.1 Software products can not meet security requirements	SEI88a,#33 Determine policy for certification of reusable components SEI88a,#35 Determine system for handling classified reusable components	
E.2.8.2 Technology for secure distributed systems and new evolving architectures not available		OSD,#28 NSA's Endorsed Products List identifies manufacturers of secure systems
E.2.8.3 Security capabilities are very costly		OSD,#28 NSA's Endorsed Products List identifies manufacturers of secure systems

Table E.2
E.2.9 Technology Base

Issues	Recommendations	Action Taken
E.2.9.1 University research base in need of considerable enhancement in the areas of faculty, equipment, facilities, and support	DSB81,#3 Increase and facilitate support for university research DSB81,#5 Upgrade computer resources in universities	Army,#23 AIRMICS effort to leverage academic RDTE
E.2.9.2 Key technology areas lack adequate research funding	DSB81,#1 Formulate integrated programs in AI and Fast Algorithms DOD82,#2i Encourage DoD personnel to review industry IR&D in software USAF83,#8 Increase funding for long range software technology research DSB87,#11 Focus research on SDI-specific objectives ZRAKET88,#13 Create a unified DoD Software Technology Plan AMC89,#1 Develop commercial product testbed, tech. R&D, and focus on reuse AMC89,#2 Develop plan, R&D, and policy for software reuse IDA89,#2 Continue funding for Service Laboratories and the DoD Software AFSB89,#3 Fund parallel programs at system level to foster wider industrial base AFSB89,#24 Increase technology base investment in software engineering	OSD,#15 STARS Program was initiated in November, 1984 OSD,#19 *ATIP established*

ANNEX F

References and Bibliography

February 9, 1990

ANNEX F

F. References and Bibliography

F.1 References

[AFSB89] *Adapting Software Development Policies to Modern Technology,* Air Force Studies Board, 1989.

[AMC89] *Report of the AMC Software Task Force,* 28 February 1989.

[BOARD88] *Ada Board Response to the Report of the Defense Science Board Task Force on Military Software,* Office of the Under Secretary of Defense for Acquisition,February 1988.

[CODSIA84] *DoD Management of Mission-Critical Computer Resources, Vols. I and II,* Council of Defense and Space Industry Associations, Report 13-82, March 1984.

[DoD82] *Report of the DoD Task Force on Software Problems,* Department of Defense, 30 July 1982.

[DSB81] *Report of the Defense Science Board 1981 Summer Study Panel on Technology Base,* November 1981.

[DSB87] *Report of the Defense Science Board Task Force on Military Software,* September 1987.

[Firth87] Firth, Robert, Vicky Mosley, Richard Pethia, Lauren Roberts, William Wood, *A Guide to the Classification and Assessment of Software Engineering Tools,* Software Engineering Institute, CMU/SEI-87-TR-10.

[IDA89] *Software Technology Development and Deployment Plan for the DoD Technology Base,* IDA Memorandum Report M-552, April 1989.

[OIG88] *Summary Report on the Defense-Wide Audit of Support for Tactical Software,* Office of the Inspector General, 7 April 1988.

[OSD89] *DEFENSE MANAGEMENT Report to the President,* by Secretary of Defense Dick Cheney, July, 1989.

[OTA89] *Holding the Edge: Maintaining the Defense Technology Base,* Congress of the United States Office of Technology Assessment, April 1989.

[SEI88a] *Proceedings of the Workshop on Executive Software Issues,* Software Engineering Institute, 1988.

[SEI88b] *SEI Software Problems Report,* DoD Strategic Planning Workshop for Software, 6 December 1988.

[USAF83] *Report of the USAF Scientific Advisory Board Ad Hoc Committee on The High Cost and Risk of Mission-Critical Software,* December 1983.

[Weiderman87] Weiderman, Habermann, Borger, Klein, Landherr, Smeston, Altman, D'Ippolito, Kochmar, Sun, *Evaluation of Ada Environments.* Software Engineering Institute, CMU/SEI-87-TR-1.

[ZRAKET88] *Report of Workshop on Military Software,* 1 July 1988, Charles A. Zraket, Chair.

F.2 Bibliography

The Armed Forces Communications and Electronics Association, *Command & Control System Acquisition Study,* July 16, 1982.

Air Force Logistics Command, *A Study Of Embedded Computer Systems Support,* September, 1980.

Betti, John A., Under Secretary of Defnse for Acquisition, "The Acquisition Process", *Defense 89,* November/December, pp. 8-12.

Brooks, Frederick P. Jr., "Nc Silver Bullet", *Computer,* April 1987, pp 10-19.

Branscomb, Anne W., *Who Owns Creativity? Property Rights in the Information Age,* Technology Review, May/June 1988.

Executive Office of the President, Office of Science and Technology Policy, *The Federal High Performance Computing Program,* September 8, 1989.

Hefley, William E., Capt. Benita L. Gilliard, CMSgt. Charles M. Worm, and Landon O. Sager, *Issues in Software: A Blue Two Visit Feasibility Assessment,* Software Engineering Institute, CMU/SEI-87-TR-42, January 1988.

Institute for Defense Analyses, *Comparison of Software Problems 1974-1988,* Memorandum Report M-508, Draft 18 October 1988.

Institute for Defense Analyses, *Report of the Task Force for Improved Coordination of the DoD Science and Technology Program, Volumes I and II,* Report R-345, August 1988.

Institute for Electrical and Electronic Engineers *Glossary of Software Engineering Terminology,* Currently in draft, to be released March 1990.

Inspector General, Department of Defense, *Report on Information Resources Management (IRM) within DoD,* Report No. 89-INS-03, February 1989.

Joint Logistics Commanders, *Orlando II Solving the PDSS Challenge,* 4th Bienniel Software Workshop, May 1987.

Martin, Anne C. and Kevin Deasy, *Seeking the Balance Between Government and Industry Interests in Software Acquisitions, Volume I: A Basis for Reconciling DoD and Industry Needs for Rights in Software,* CMU/SEI-87-TR-13, June 1987.

National Research Council, *Air force Base Level Automation Environment: Final Report to the Department of the Air Force,* National Academy Press, 1987.

National Research Council, Computer Science and Technology Board, Commission on Physical Sciences, Mathematics, and Resources, *The National Challenge in Computer Science and Technology,* National Academy Press, 1988.

National Research Council, Computer Science and Technology Board, Commission on Physical Sciences, Mathematics, and Resources, *Scaling Up: A Research Agenda For Software Engineering,* National Academy Press, 1989.

National Science Foundation, *Towards a National Collaboratory,* Report of an Invitational Workshop, March 17-18, 1989.

Samuelson, Pamela, *Adequate Planning for Acquiring Sufficient Documentation about ·nd Rights in Software to Permit Organic or Competitive Maintenance,* CMU/SEI-86-TM-1, March 1986.

Samuelson, Pamela, *Understanding the Implications of Selling Rights in Software to the Defense Department: A Journey through the Regulatory Maze,* CMU/SEI-86-TM-3, March 1986.

Samuelson, Pamela, *Toward a Reform of the Defense Department Software Acquisition Policy,* CMU/SEI-86-TR-1, April 1986.

Society for Industrial and Applied Mathematics, *A National Computing Initiative: The Agenda for Leadership,* Report of the Panel on Research Issues in Large-Scale Computational Science and Engineering, Workshop Report, February, 1987.

Tenenbaum, J., *The Military Computing Crisis: The Search for a Solution: Summary and Investment Opinion,* Salamon Brothers, Inc., 22 September 1987.

U.S. Department of Defense *Software Technology for Adaptable, Reliable Systems (STARS),* Joint Task Force Report, 15 March 1983.

U. S. Air Force, *Software Management Action Plan (Project Bold Stroke),*, 29 November 1985.

U. S. Congress, Committees on Armed Services, *The Department of Defense Critical Technologies Plan,* March 15, 1989.

U.S. House of Representatives, Committee on Science, Space, and Technology, *Bugs in the Program: Problems in Federal Government Computer Software Development and Regulation,* August 3, 1989.

Weiderman, Nelson H., Mark W. Borger, Andrea L. Cappellini, Susan A. Dart, Mark H. Klein, and Stefan F. Landherr, *Ada for Embedded Systems: Issues and Questions,* Software Engineering Institute, CMU/SEI-87-TR-26, December 1987.

ANNEX G

List of Acronyms

February 9, 1990

ANNEX G

G. List of Acronyms

ACAT	Acquisition Category
ACEC	Ada Compiler Evaluation Capability
ADP	Automated Data Processing
ADPE	Automatic Data Processing Equipment
AF	Air Force
AFATL	Air Force Armament Laboratory
AFB	Air Force Base
AFLC	Air Force Logistics Command
AFR	Air Force Regulation
AFSC	Air Force Systems Command
AFSCP	Air Force Systems Command Pamphlet
AIMI	AMPS Intelligent Multimedia Interface
AIRMICS	Army Institute for Research Management of Information, Communications, and Computer Sciences
AIS	Automated Information Systems
AJPO	Ada Joint Program Office
ALBM	Air—Land Battle Management
ALC	Air Logistics Center
ALS/N	Ada Language System/Navy
AMC	Army Materiel Command or Armament, Munitions, and Chemical
AMCCOM	(Army) Armament, Munitions and Chemical Command
AMPS	Advanced Mission Planning System
ANSI	American National Standards Institute
APMS	*Automated Project Management System*
APSE	Ada Programming Support Environment
AR	Army Regulation
ASN	Assistant Secretary of the Navy
ASOS	Army Secure Operating System
ASQS	Assistant for Specifying the Quality of Software
ATCCS	Army Tactical Command and Control System
ATE	Automatic Test Equipment
ATIP	Ada Technology Insertion Program
ATN	Adaptive Tactical Navigation
ATVS	Ada Test and Verification System
BM	Battle Management
BaRT	Bayesian Reasoning Tool
C^2I	Command, Control, and Intelligence
C^3I	Command, Control, Communications, and Intelligence
C^4I	Command, Control, Communications, Computer, and Intelligence
C^4I^2	Command, Control, Communications, Computer, Intelligence, and Interoperability
CAD	Computer Aided Design
CAE	Computer Aided Engineering
CAIS	Common APSE Interface Set
CALS	Computer Aid for Acquisition and Logistics Systems
CAMP	Common Ada Missile Package
CAMPS	Core of A Mission Planning System
CASE	Computer Aided Software Engineering
CECOM	(Army) Communications - Electronics Command
CNET	Chief, Naval Education & Training
CNETINST	CNET Instruction
COCOMO	Constructive Cost Model
COTS	Commercial Off-The-Shelf
CRFP	Computer Resources Focal Point
CRLCMP	Computer Resources Life Cycle Management Plan
CRMP	Computer Resources Management Plan
CRWG	Computer Resources Working Group
CSCI	Computer Software Configuration Item
DAB	Defense Acquisition Board
DAR	Defense Acquisition Regulation
DARPA	Defense Advanced Research Projects Agency
Date	Decision Aids Test Environment
DBMS	Database Management System
DCA	Defense Communications Agency
DCDS	Distributed Computing Design System
DDDRE	Deputy Director of Defense Research and Engineering
DDN	Defense Data Network
DIDS	Defense Integrated Data System
DIF	Document Interchange Format
DLA	Defense Logistics Agency
DoD, DOD	Department of Defense
DOD-STD	Department of Defense Standard
DOE	Department of Energy
DONIRM	Department of the Navy Information Resources Management
DSMC	Defense Systems Management College
DSS	Decision Support Shell
ECP	Engineering Change Proposal
ECR	Embedded Computer Resources
ECS	Embedded Computer System or Engineering Change System
EDI	Electronic Data Interchange
EFG	Evidence Flow Graph
ESIP	Embedded Computer Resources Support Improvement Program
ESKIT	(DARPA) Experimental System Kit
EVPA	Experimental Version Performance Assessment
FAR	Federal Acquisition Regulation
FCA	Functional Configuration Audit
FCTC	Federal Compiler Testing Center
FDS	(Navy) Fixed Distributed System
FED-STD	Federal Telecommunications Standard
FFIT	Form-Fit-Interoperability-Transportability
FIPS	Federal Information Processing Standards
FIRMR	Federal Information Resources Mangement Regulations
FITS	Fault Identification & Test Subsystems
FPMR	Federal Property Management Regulation
FSD	Full Scale Development
FTP	File Transfer Protocol
FTPP	Fault Tolerant Parallel Processor
FY	Fiscal Year

GOSIP	Government Open Systems Interconnect Profile
GSA	General Services Administration
HFE	Human Factors Engineering
HGI	(Navy) High Gain Initiative
HOL	Higher Order Language
HPC	(Navy) High Performance Computing
ICBM	Intercontinental Ballistic Missiles (ICBM PO - Program Office)
IS	Information System
ISDN	Integrated Services Digital Network
ISEB	Information System Executive Board
ISO	International Standards Organization
IV&V	Independent Verification and Validation
JIAWG	Joint Integrated Avionics Working Group
JLC	Joint Logistics Command/Commanders
JSTPS	Joint Strategic Planning Staff
KBES	Knowledge Base Execution System
KBSA	Knowledge-Based Software Assistant
LCSA	Life Cycle Support Agent
LCSE	Life Cycle Support Environment
LCSEC	Life Cycle Software Engineering Steering Committee
LSA	Logistic Support Analysis
LSAR	Logistic Support Analysis Record
LSQE	Language Sensitive Quality Editor
MAC	Military Airlift Command
MAISRC	Major Automated Information Systems Review Council
MCCDPA	Marine Corps Central Design and Programming Activities
MCCR	Mission-Critical Computer Resources
MCDS	Mission Critical Defense System
MCO	Marine Corps Order
MCPDM	Marine Corps Program Decision Meetings
MCRDAC	Marine Corps Research, Development, & Acquisition Command
MECS	Modular Embedded Computer Software
MIL-STD	Military Standard
MLS	Multilevel Secure
MMI	Man—Machine Interface
MSCR	Materiel Systems Computer Resources
NADC	Naval Air Development Center
NASA	National Aeronautics and Space Administation
NATO	North Atlantic Treaty Organization
NAVAIR	Naval Air Systems Command
NAVAIRINST	NAVAIR Instruction
NAVMAT	Naval Material Command
NAVMATINST	Naval Material Command Instruction
NAVSEA	Naval Sea Systems Command
NAVSEAINST	Naval Sea Systems Command Instruction
NDI	Non-Developmental Item
NGCR	Next Generation Computer Resources
NIST	National Institute of Standards and Technology
NOSC	Naval Ocean Systems Center
NRL	Navy Research Lab
NSA	National Security Agency
NSF	National Science Foundation
NTB	National Test Bed
NTDS	Naval Tactical Data System
NTF	National Test Facility
NWC	Naval Weapons Center
OASD	Office of the Assistant Secretary of Defense
OMB	Office of Management and Budget
OMINST	Operations and Maintenance Instruction
ONR	Office of Naval Research
OOD	Object Oriented Design
OPNAV	Office of the Chief of Naval Operations
OPNAVINST	OPNAV Instruction
OPR	Office of Primary Responsibility
OSD	Office of the Secretary of Defense
OSS	(Navy) Operation Support System
OUSD	Office of the Under Secretary for Defense
P&L	Production & Logistics
PDSS	Post Deployment Software Support
PE	Program Element
PEEP	Parallel Evaluation and Experimentation Platform
PEO	Program Executive Officer
PM	Program/Project/Product Manager
PMA	Project Management Assistant
PPBS	Programming, Planning, and Budgeting System
PRIMO	Plausible Reasoning Module
PSE	Programming Support Environment
PSL	Programming Support Language
PSN	Packet Switch Nodes
QUES	Quality Evaluation System
R&AT	Research and Advanced Technology
RADC	Rome Air Development Center
RDT&E	Research, Development, Test and Evaluation
RISC	Reduced Instruction Set Computer
RPM	Reliability Prediction Model
S&T	Science and Technology
SAC	Strategic Air Command
SAM	Super Abstract Machine
SAME	SQL Ada Module Extensions
SAPE	Survivable Adaptive Planning Experiment
SBIR	Small Business Innovative Research
SCI	Sensitive Compartmented Information
SCRB	Software Change Review Boards
SDCCR	Software Development Capability/Capacity Review
SDDS	Strategic Defense Development System
SDI	Strategic Defense Initiative
SDIO	Strategic Defense Initiative Organization
SDIP	Software Development Integrity Program
SDM	System Description Model
SDP	System Decision Paper or Software Development Plan
SDS	Strategic Defense System
SE	Software Engineering
SECNAV	Secretary of the Navy
SECNAVINST	Secretary of the Navy Instruction
SECNAVNOTE	Secretary of the Navy Note
SECOMO	Software Engineering Cost Model
SECR	Standard Embedded Computer Resources

SEE	Software Engineering Environment
SEI	Software Engineering Institute
SLCMP	Software Life-Cycle Management Plan
SLCSE	Software Life Cycle Support Environment
SPAWAR	Space and Naval Warfare Systems Command
SPAWARINST	Space and Naval Warfare Systems Command Instruction
SPC	Software Productivity Consortium
SPMS	SLCSE Project Management System
SQL	Structured Query Language
SSDL	SDDS System Design Language
SSPRM	(Marine Corps) Software Support Personnel Model
STARS	Software Technology for Adaptable, Reliable Systems
STOCS	Synchronous Token-Based Communication State
SWFT	Software Fault Tolerant
T&E	Test and Evaluation
TAC	Technology Assessment Center or Tactical Air Command
TADSTAND	Navy Tactical Data Standard
TCSEC	Trusted Computer System Evaluation Criteria
TD	(Navy) Technical Directive
TDBMS	Trusted Database Management System
TEMP	Test and Evaluation Master Plan
TESSE	Test Environment Support and System Enhancements
TRADOC	(Army) Training and Doctrine Command
TSP	Teleprocessing Service Program or (SDI) Trusted Software Project
USAF	United States Air Force
USTRANSCOM	United States Transportation Command
WIS	WWMCCS Information System
WRDC	Wright Research and Development Center
WWMCCS	World-Wide Military Command and Control System